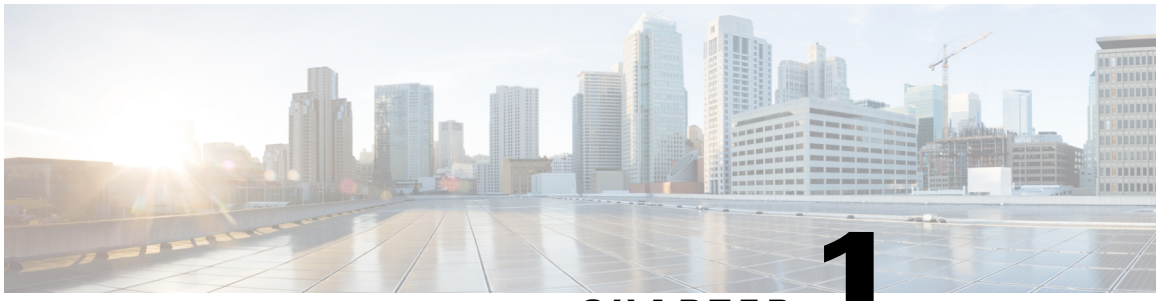
User Security Configuration Guide, Cisco IOS Release 15SY

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

CONTENTS



CHAPTER 1

Cisco IOS Login Enhancements-Login Block

The Cisco IOS Login Enhancements (Login Block) feature allows users to enhance the security of a router by configuring options to automatically block further login attempts when a possible denial-of-service (DoS) attack is detected.

The login block and login delay options introduced by this feature can be configured for Telnet or SSH virtual connections. By enabling this feature, you can slow down “dictionary attacks” by enforcing a “quiet period” if multiple failed connection attempts are detected, thereby protecting the routing device from a type of denial-of-service attack.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About Cisco IOS Login Enhancements

Protecting Against Denial of Service and Dictionary Login Attacks

Connecting to a routing device for the purposes of administering (managing) the device, at either the User or Executive level, is most frequently performed using Telnet or SSH (secure shell) from a remote console (such as a PC). SSH provides a more secure connection option because communication traffic between the user's device and the managed device are encrypted. The Login Block capability, when enabled, applies to both Telnet connections and SSH connections. Beginning in Release versions 12.3(33)SRB2, 12.2(33)SXH2, and 12.4(15)T1, the Login Block capability also applies to HTTP connections."

The automated activation and logging of the Login Block and Quiet Period capabilities introduced by this feature are designed to further enhance the security of your devices by specifically addressing two well known methods that individuals use to attempt to disrupt or compromise network devices.

If the connection address of a device is discovered and is reachable, a malicious user may attempt to interfere with the normal operations of the device by flooding it with connection requests. This type of attack is referred to as an attempted Denial-of-Service, because it is possible that the device may become too busy trying to process the repeated login connection attempts to properly handle normal routing services or are not able to provide the normal login service to legitimate system administrators.

The primary intention of a dictionary attack, unlike a typical DoS attack, is to actually gain administrative access to the device. A dictionary attack is an automated process to attempt to login by attempting thousands, or even millions, of username/password combinations. (This type of attack is called a "dictionary attack" because it typically uses, as a start, every word found in a typical dictionary as a possible password.) As scripts or programs are used to attempt this access, the profile for such attempts is typically the same as for DoS attempts; multiple login attempts in a short period of time.

By enabling a detection profile, the routing device can be configured to react to repeated failed login attempts by refusing further connection request (login blocking). This block can be configured for a period of time, called a "quiet period". Legitimate connection attempts can still be permitted during a quiet period by configuring an access-list (ACL) with the addresses that you know to be associated with system administrators.

Login Enhancements Functionality Overview

Delays Between Successive Login Attempts

A Cisco IOS device can accept virtual connections as fast as they can be processed. Introducing a delay between login attempts helps to protect the Cisco IOS software-based device against malicious login connections such as dictionary attacks and DoS attacks. Delays can be enabled in one of the following ways:

- Through the **auto secure** command. If you enable the AutoSecure feature, the default login delay time of one second is automatically enforced.
- Through the **login block-for** command. You must enter this command before issuing the **login delay** command. If you enter only the **login block-for** command, the default login delay time of one second is automatically enforced.

- Through the new global configuration mode command, **login delay**, which allows you to specify login delay time to be enforced, in seconds.

Login Shutdown If DoS Attacks Are Suspected

If the configured number of connection attempts fail within a specified time period, the Cisco IOS device does not accept any additional connections for a "quiet period." (Hosts that are permitted by a predefined access-control list [ACL] are excluded from the quiet period.)

The number of failed connection attempts that trigger the quiet period can be specified through the new global configuration mode command **login block-for**. The predefined ACL that is excluded from the quiet period can be specified through the new global configuration mode command **login quiet-mode access-class**.

This functionality is disabled by default, and it is not enabled if AutoSecure if enabled.

How to Configure Cisco IOS Login Enhancements

Configuring Login Parameters

Use this task to configure your Cisco IOS device for login parameters that help detect suspected DoS attacks and slow down dictionary attacks.

All login parameters are disabled by default. You must issue the **login block-for** command, which enables default login functionality, before using any other login commands. After the **login block-for** command is enabled, the following defaults are enforced:

- A default login delay of one second
- All login attempts made through Telnet or SSH are denied during the quiet period; that is, no ACLs are exempt from the login period until the **login quiet-mode access-class** command is issued.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **login block-for** *seconds* **attempts** *tries* **within** *seconds*
4. **login quiet-mode access-class** {*acl-name* | *acl-number*}
5. **login delay** *seconds*
6. **exit**
7. **show login failures**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: `Router> enable`	• Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**login block-for** *seconds* **attempts** *tries* **within** *seconds* **Example:** `Router(config)# login block-for 100 attempts 2 within 100`	Configures your Cisco IOS device for login parameters that help provide DoS detection. **Note** This command must be issued before any other login command can be used.

Step 4	**login quiet-mode access-class** {*acl-name* | *acl-number*}
	Example:
	`Router(config)# login quiet-mode access-class myacl`
		(Optional) Although this command is optional, it is recommended that it be configured to specify an ACL that is to be applied to the router when the router switches to quiet mode. When the router is in quiet mode, all login requests are denied and the only available connection is through the console.
		If this command is not configured, then the default ACL **sl_def_acl** is created on the router. This ACL is hidden in the running configuration. Use the **show access-list sl_def_acl** to view the parameters for the default ACL.
		For example:
		`Router#show access-lists sl_def_acl`
		`Extended IP access list sl_def_acl`
		`10 deny tcp any any eq telnet`
		`20 deny tcp any any eq www`
		`30 deny tcp any any eq 22`
		`40 permit ip any any`

Step 5	**login delay** *seconds* **Example:** `Router(config)# login delay 10`	(Optional) Configures a delay between successive login attempts.

	Command or Action	Purpose
Step 6	**exit** **Example:** `Router# exit`	Exits to privileged EXEC mode.
Step 7	**show login failures** **Example:** `Router# show login`	Displays login parameters. • **failures** --Displays information related only to failed login attempts.

Configuration Examples for Login Parameters

Setting Login Parameters Example

The following example shows how to configure your router to enter a 100 second quiet period if 15 failed login attempts is exceeded within 100 seconds; all login requests are denied during the quiet period except hosts from the ACL "myacl."

```
Router(config)# login block-for 100 attempts 15 within 100
Router(config)# login quiet-mode access-class myacl
```

Showing login Parameters Example

The following sample output from the **show login** command verifies that no login parameters have been specified:

```
Router# show login
No login delay has been applied.
No Quiet-Mode access list has been configured.
All successful login is logged and generate SNMP traps.
All failed login is logged and generate SNMP traps
Router NOT enabled to watch for login Attacks
```

The following sample output from the **show login** command verifies that the **login block-for**command is issued. In this example, the command is configured to block login hosts for 100 seconds if 16 or more login requests fail within 100 seconds; five login requests have already failed.

```
Router# show login
A default login delay of 1 seconds is applied.
No Quiet-Mode access list has been configured.
All successful login is logged and generate SNMP traps.
All failed login is logged and generate SNMP traps.
Router enabled to watch for login Attacks.
If more than 15 login failures occur in 100 seconds or less, logins will be disabled for
100 seconds.
```

```
Router presently in Watch-Mode, will remain in Watch-Mode for 95 seconds.
Present login failure count 5.
```

The following sample output from the **show login** command verifies that the router is in quiet mode. In this example, the **login block-for** command was configured to block login hosts for 100 seconds if 3 or more login requests fail within 100 seconds.

```
Router# show login
A default login delay of 1 seconds is applied.
No Quiet-Mode access list has been configured.
All successful login is logged and generate SNMP traps.
All failed login is logged and generate SNMP traps.
Router enabled to watch for login Attacks.
If more than 2 login failures occur in 100 seconds or less, logins will be disabled for 100
 seconds.
Router presently in Quiet-Mode, will remain in Quiet-Mode for 93 seconds.
Denying logins from all sources.
```

The following sample output from **show login failures** command shows all failed login attempts on the router:

```
Router# show login failures
Information about login failure's with the device
Username      Source IPAddr  lPort Count  TimeStamp
try1          10.1.1.1        23    1     21:52:49 UTC Sun Mar 9 2003
try2          10.1.1.2        23    1     21:52:52 UTC Sun Mar 9 2003
```

The following sample output from **show login failures** command verifies that no information is presently logged:

```
Router# show login failures
*** No logged failed login attempts with the device.***
```

Additional References

Related Documents

Related Topic	Document Title
AutoSecure	AutoSecure feature module.
Secure Management/Administrative Access	Role-Based CLI Access feature module.

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

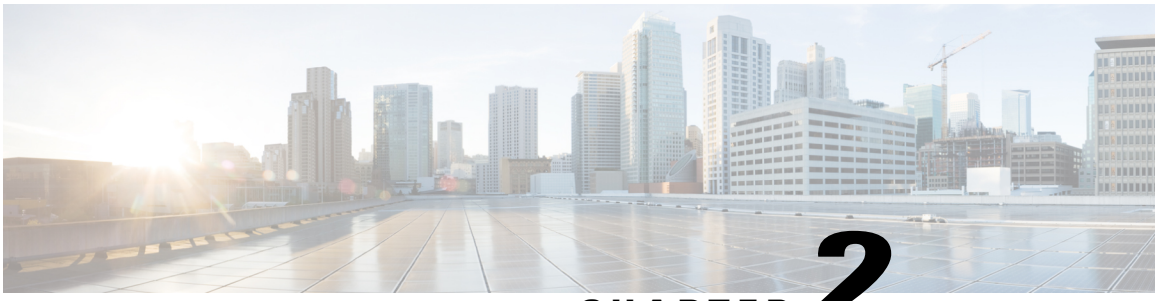
Feature Information for Cisco IOS Login Enhancements-Login Block

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 1: Feature Information for Cisco IOS Login Enhancements (Login Block)

Feature Name	Releases	Feature Information
Cisco IOS Login Enhancements (Login Block)	12.3(4)T 12.2(25)S 12.2(33)SRA 12.2(33)SRB 12.2(33)SXH 12.4(15)T1	The Cisco IOS Login Enhancements (Login Block) feature allows users to enhance the security of a router by configuring options to automatically block further login attempts when a possible DoS attack is detected. This feature was introduced in Cisco IOS Release 12.3(4)T. This feature was integrated into Cisco IOS Release 12.2(25)S. This feature was integrated into Cisco IOS Release 12.2(33)SRA. Support for HTTP login blocking was integrated into Cisco IOS Release 12.2(33)SRB, 12.2(33)SXH, 12.4(15)T1.



CHAPTER 2

Configuring Security with Passwords, Privileges, and Logins

Cisco IOS software-based networking devices provide several features that can be used to implement basic security for command-line sessions using only the operating system running on the device. These features include the following:

- Different levels of authorization for CLI sessions to control access to commands that can modify the status of the networking device versus commands that are used to monitor the device
- Assigning passwords to CLI sessions
- Requiring users to log in to a networking device with a username
- Changing the privilege levels of commands to create new authorization levels for CLI sessions

This module is a guide to implementing a baseline level of security for your networking devices. It focuses on the least complex options available for implementing a baseline level of security. If you have networking devices installed in your network with no security options configured, or you are about to install a networking device and you need help understanding how to implement a baseline of security, this document will help you.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for Configuring Security with Passwords, Privileges, and Logins

Your networking device must not be configured to use any local or remote authentication, authorization, and accounting (AAA) security features. This document describes only the non-AAA security features that can be configured locally on the networking device.

For information on how to configure AAA security features that can be run locally on a networking device, or for information on how to configure remote AAA security using TACACS+ or RADIUS servers, see the *Securing User Services Configuration Guide Library* .

Information About Configuring Security with Passwords, Privileges, and Logins

Benefits of Creating a Security Scheme

The foundation of a good security scheme in the network is the protection of the user interfaces of the networking devices from unauthorized access. Protecting access to the user interfaces on your networking devices prevents unauthorized users from making configuration changes that can disrupt the stability of your network or compromise your network security.

The features described in this document can be combined in many different ways to create a unique security scheme for each of your networking devices.

You can enable nonadministrative users to run a subset of the administrative commands available on the networking device by lowering the entitlement level for the commands to the nonadministrative privilege level. This can be useful for the following scenarios:

Cisco IOS CLI Modes

To aid in the configuration of Cisco devices, the Cisco IOS CLI is divided into different command modes. Each command mode has its own set of commands available for the configuration, maintenance, and monitoring of device and network operations. The commands available to you at any given time depending on the mode

you are in. Entering a question mark (**?**) at the system prompt (device prompt) allows you to obtain a list of commands available for each command mode.

The use of specific commands allows you to navigate from one command mode to another. The standard order in which a user would access the modes is as follows: user EXEC mode; privileged EXEC mode; global configuration mode; specific configuration modes; configuration submodes; and configuration subsubmodes.

Note The default configuration of a Cisco IOS software-based networking device allows you to configure passwords to protect access only to user EXEC mode (for local and remote CLI sessions) and privileged EXEC mode. This document describes how you can provide additional levels of security by protecting access to other modes, and commands, using a combination of usernames, passwords and the **privilege** command.

Most EXEC mode commands are one-time commands, such as **show** or **more** commands, which show the current configuration status, and **clear** commands, which clear counters or interfaces. EXEC mode commands are not saved across reboots of the device.

From privileged EXEC mode, you can enter global configuration mode. In this mode, you can enter commands that configure general system characteristics. You also can use global configuration mode to enter specific configuration modes. Configuration modes, including global configuration mode, allow you to make changes to the running configuration. If you later save the configuration, these commands are stored across device reboots.

From global configuration mode you can enter a variety of protocol-specific or feature-specific configuration modes. The CLI hierarchy requires that you enter these specific configuration modes only through global configuration mode. For example, interface configuration mode is a commonly used configuration mode.

From configuration modes, you can enter configuration submodes. Configuration submodes are used for the configuration of specific features within the scope of a given configuration mode. For example, the subinterface configuration mode is a submode of the interface configuration mode.

ROM monitor mode is a separate mode used when the device cannot boot properly. If your system (router, switch, or access server) does not find a valid system image to load when it is booting, the system will enter ROM monitor mode. ROM monitor (ROMMON) mode can also be accessed by interrupting the boot sequence during startup.

The following sections contain detailed information on these command modes.

User EXEC Mode

When you start a session on a device, you generally begin in user EXEC mode, which is one of two access levels of the EXEC mode. For security purposes, only a limited subset of EXEC commands are available in user EXEC mode. This level of access is reserved for tasks that do not change the configuration of the device, such as determining the device status.

If your device is configured to require users to log in the login process will require a username and a password. If you enter incorrect password three times, the connection attempt is refused.

User EXEC mode is set by default to privilege level 1. Privileged EXEC mode is set by default to privilege level 15. For more information see the Privileged EXEC Mode, on page 12. When you are logged in to a networking device in user EXEC mode your session is running at privilege level 1. When you are logged in to a networking device in privileged EXEC mode your session is running at privilege level 15. You can move commands to any privilege level between 1 and 15 using the **privilege** command. See the Cisco IOS Privilege Levels, on page 17 for more information on privilege levels and the **privilege** command.

In general, the user EXEC commands allow you to connect to remote devices, change terminal line settings on a temporary basis, perform basic tests, and list system information.

To list the available user EXEC commands, enter a question mark (**?**). The list of commands will vary depending on the software feature set and platform you are using.

The user EXEC mode prompt consists of the hostname of the device followed by an angle bracket (>), for example, Device>.

The default hostname is generally Device, unless it has been changed during initial configuration using the **setup** EXEC command. You can also change the hostname using the **hostname** global configuration command.

Note Examples in Cisco IOS documentation assume the use of the default name of "Device." Different devices (for example, access servers) may use a different default name. If the device (router, access server, or switch) has been named with the **hostname** command, that name appears as the prompt instead of the default name.

Note You can enter commands in uppercase, lowercase, or mixed case. Only passwords are case-sensitive. However, Cisco IOS documentation convention is to always present commands in lowercase.

Privileged EXEC Mode

In order to have access to all commands, you must enter privileged EXEC mode, which is the second level of access for the EXEC mode. Normally, you must enter a password to enter privileged EXEC mode. In privileged EXEC mode, you can enter any EXEC command, because privileged EXEC mode is a superset of the user EXEC mode commands.

Because many privileged EXEC mode commands set operating parameters, privileged EXEC level access should be password protected to prevent unauthorized use. The privileged EXEC command set includes those commands contained in user EXEC mode. Privileged EXEC mode also provides access to configuration modes through the **configure**command, and includes advanced testing commands, such as **debug**.

Privileged EXEC mode is set by default to privilege level 15. User EXEC mode is set by default to privilege level 1. For more information see the User EXEC Mode, on page 11. By default the EXEC commands at privilege level 15 are a superset of those available at privilege level 1. You can move commands to any privilege level between 1 and 15 using the **privilege** command. See the Cisco IOS Privilege Levels, on page 17 for more information on privilege levels and the **privilege** command.

The privileged EXEC mode prompt consists of the hostname of the device followed by a pound sign (#), for example, Device#.

To access privileged EXEC mode, use the **enable** command. If a privileged EXEC mode password has been configured the system will prompt you for a password after you issue the **enable** command. Use the **exit** command to leave privileged EXEC mode.

Note Privileged EXEC mode is sometimes referred to as "enable mode," because the **enable** command is used to enter the mode.

If a password has been configured on the system, you will be prompted to enter it before being allowed access to privileged EXEC mode. The password is not displayed on the screen and is case-sensitive. If an enable password has not been set, privileged EXEC mode can be accessed only by a local CLI session (terminal connected to the console port).

If you attempt to access privileged EXEC mode on a device over a remote connection, such as aTelnet connection, and you have not configured a password for privileged EXEC mode, you will see the **% No password set** error message. For more information on remote connections see the Remote CLI Sessions, on page 15. The system administrator uses the **enable secret** or **enable password**global configuration command to set the password that restricts access to privileged EXEC mode. For information on configuring a password for privileged EXEC mode, see the Protecting Access to Privileged EXEC Mode, on page 26.

To return to user EXEC mode, use the **disable** command:

Note that the password will not be displayed as you type, but is shown here for illustrational purposes. To list the commands available in privileged EXEC mode, issue question mark (**?**) at the prompt. From privileged EXEC mode you can access global configuration mode, which is described in the following section.

Note Because the privileged EXEC command set contains all of the commands available in user EXEC mode, some commands can be entered in either mode. In Cisco IOS documentation, commands that can be entered in either user EXEC mode or privileged EXEC mode are referred to as EXEC mode commands. If user or privileged is not specified in the documentation, assume that you can enter the referenced commands in either mode.

Global Configuration Mode

The term “global” is used to indicate characteristics or features that affect the system as a whole. Global configuration mode is used to configure your system globally, or to enter specific configuration modes to configure specific elements such as interfaces or protocols. Use the **configure terminal**privileged EXEC command to enter global configuration mode.

To access global configuration mode, use the **configure terminal** command in privileged EXEC mode:

Note that the system prompt changes to indicate that you are now in global configuration mode. The prompt for global configuration mode consists of the hostname of the device followed by (config) and the pound sign (#). To list the commands available in privileged EXEC mode, issue **?** at the prompt.

Commands entered in global configuration mode update the running configuration file as soon as they are entered. In other words, changes to the configuration take effect each time you press the Enter or Return key at the end of a valid command. However, these changes are not saved into the startup configuration file until you issue the **copy running-config startup-config** EXEC mode command.

The system dialog prompts you to end your configuration session (exit configuration mode) by pressing the Control (Ctrl) and “z” keys simultaneously; when you press these keys, **^Z** is printed to the screen. You can actually end your configuration session by entering the Ctrl-Z key combination, using the **end** command, and using the Ctrl-C key combination. The **end** command is the recommended way to indicate to the system that you are done with the current configuration session.

Caution If you use Ctrl-Z at the end of a command line in which a valid command has been typed, that command will be added to the running configuration file. In other words, using Ctrl-Z is equivalent to hitting the Enter (Carriage Return) key before exiting. For this reason, it is safer to end your configuration session using the **end** command. Alternatively, you can use the Ctrl-C key combination to end your configuration session without sending a Carriage Return signal.

You can also use the **exit** command to return from global configuration mode to EXEC mode, but this only works in global configuration mode. Pressing Ctrl-Z or entering the **end** command will always take you back to EXEC mode regardless of which configuration mode or configuration submode you are in.

To exit global configuration command mode and return to privileged EXEC mode, use the **end** or **exit** command.

From global configuration mode, you can enter a number of protocol-specific, platform-specific, and feature-specific configuration modes.

Interface configuration mode, described in the following section, is an example of a configuration mode you can enter from global configuration mode.

Interface Configuration Mode

One example of a specific configuration mode you can enter from global configuration mode is interface configuration mode.

Many features are enabled on a per-interface basis. Interface configuration commands modify the operation of an interface such as an Ethernet, FDDI, or serial port. Interface configuration commands always follow an **interface** global configuration command, which defines the interface type.

For details on interface configuration commands that affect general interface parameters, such as bandwidth or clock rate, refer to the *Cisco IOS Interface Configuration Guide* .

To access and list the interface configuration commands, use the interface type number command.

To exit interface configuration mode and return to global configuration mode, enter the **exit** command.

Configuration submodes are configuration modes entered from other configuration modes (besides global configuration mode). Configuration submodes are for the configuration of specific elements within the configuration mode. One example of a configuration submode is subinterface configuration mode, described in the following section.

Subinterface Configuration Mode

From interface configuration mode, you can enter subinterface configuration mode. Subinterface configuration mode is a submode of interface configuration mode. In subinterface configuration mode you can configure multiple virtual interfaces (called subinterfaces) on a single physical interface. Subinterfaces appear to be distinct physical interfaces to the various protocols.

For detailed information on how to configure subinterfaces, refer to the appropriate documentation module for a specific protocol in the Cisco IOS software documentation set.

To exit subinterface configuration mode and return to interface configuration mode, use the **exit** command. To end your configuration session and return to privileged EXEC mode, press Ctrl-Z or enter the **end** command.

Cisco IOS CLI Sessions

Local CLI Sessions

Local CLI sessions require direct access to the console port of the networking device. Local CLI sessions start in user EXEC mode. See the Cisco IOS CLI Modes, on page 10 for more information on the different modes that are supported on your networking device. All of the tasks required to configure and manage a networking device can be done using a local CLI session. The most common method for establishing a local CLI session is to connect the serial port on a PC to the console port of the networking device and then to launch a terminal emulation application on the PC. The type of cable and connectors required and the settings for the terminal emulation application on the PC depend on the type of networking device that you are configuring. See the documentation for your networking device for more information on setting it up for a local CLI session.

Remote CLI Sessions

Remote CLI sessions are created between a host such as a PC and a networking device such as a router over a network using a remote terminal access application such as Telnet and SSH. Local CLI sessions start in user EXEC mode. See the Cisco IOS CLI Modes, on page 10 for more information on the different modes that are supported on your networking device. Most of the tasks required to configure and manage a networking device can be done using a remote CLI session. The exceptions are tasks that interact directly with the console port (such as recovering from a corrupted operating system (OS) by uploading a new OS image over the console port) and interacting with the networking device when it is in ROMMON mode.

Telnet is the most common method for accessing a remote CLI session on a networking device.

Note SSH is a more secure alternative to Telnet. SSH provides encryption for the session traffic between your local management device such as a PC and the networking device that you are managing. Encrypting the session traffic with SSH prevents hackers that might intercept the traffic from being able to decode it. See Secure Shell Version 2 Support feature module for more information on using SSH.

Terminal Lines Used for Local and Remote CLI Sessions

Cisco networking devices use the word “lines” to refer to the software components that manage local and remote CLI sessions. You use the **line console 0** global configuration command to enter line configuration mode to configure options, such as a password, for the console port.

Remote CLI sessions use lines that are referred to as vty lines. You use the **line vty** *line-number*[*ending-line-number*]global configuration command to enter line configuration mode to configure options, such as a password, for remote CLI sessions.

Protection of Access to Cisco IOS EXEC Modes

Protection of Access to User EXEC Mode

The first step in creating a secure environment for your networking device is protecting access to user EXEC mode by configuring passwords for local and remote CLI sessions.

You can protect access to user EXEC mode for local CLI sessions by configuring a password on the console port. See the Configuring a Password for Local CLI Sessions, on page 24.

You can protect access to user EXEC mode for remote CLI sessions by configuring a password on the vtys. See the Configuring a Password for Remote CLI Sessions, on page 22 for instructions on how to configure passwords for remote CLI sessions.

Protection of Access to Privileged EXEC Mode

The second step in creating a secure environment for your networking device is protecting access to privileged EXEC mode with a password. The method for protecting access to privileged EXEC mode is the same for local and remote CLI sessions.

You can protect access to privileged EXEC mode by configuring a password for it. This is sometimes referred to as the enable password because the command to enter privileged EXEC mode is **enable**.

Cisco IOS Password Encryption Levels

Some of the passwords that you configure on your networking device are saved in the configuration in plain text. This means that if you store a copy of the configuration file on a disk, anybody with access to the disk can discover the passwords by reading the configuration file. The following password types are stored as plain text in the configuration by default:

- Console passwords for local CLI sessions.
- Virtual terminal line passwords for remote CLI sessions.
- Username passwords using the default method for configuring the password.
- Privileged EXEC mode passwords when they are configured with the **enable password** *password* command.
- Authentication key chain passwords used by Routing Information Protocol version 2 (RIPv2) and Enhanced Interior Gateway Routing Protocol (EIGRP).
- BGP passwords for authenticating BGP neighbors.
- Open Shortest Path First (OSPF) authentication keys for authenticating OSPF neighbors.
- Intermediate System-Intermediate System (IS-IS) passwords for authenticating ISIS neighbors.

The following excerpt from a router configuration file shows examples of passwords and authentication keys that are stored as clear text:

```
!
enable password O9Jb6D
```

```
!
username username1 password 0 kV9sIj3
!
key chain trees
 key 1
  key-string key1
!
interface Ethernet1/0.1
 ip address 172.16.6.1 255.255.255.0
 ip router isis
 ip rip authentication key-chain key2
 ip authentication key-chain eigrp 1 key2
 ip ospf authentication-key j7876
 no snmp trap link-status
 isis password u7865k
!
line vty 0 4
 password V9jA5M
!
```

You can encrypt these clear text passwords in the configuration file by using the **service password-encryption** command. This should be considered as a minimal level of security because the encryption algorithm used by the **service password-encryption** command to encrypt passwords creates text strings that can be decrypted using tools that are publicly available. You should still protect access to any electronic or paper copies of your configuration files after you use the **service password-encryption** command.

The **service password-encryption** command does not encrypt the passwords when they are sent to the remote device. Anybody with a network traffic analyzer who has access to you network can capture these passwords from the packets as they are transmitted between the devices. See the Configuring Password Encryption for Clear Text Passwords, on page 28 for more information on encrypting clear text passwords in configuration files.

Many of the Cisco IOS features that use clear text passwords can also be configured to use the more secure message digest algorithm 5 (MD5). The MD5 algorithm creates a text string in the configuration file that is much more difficult to decrypt. The MD5 algorithm does not send the password to the remote device. This prevents people using a traffic analyzer to capture traffic on your network from being able to discover your passwords.

You can determine the type of password encryption that has been used by the number that is stored with the password string in the configuration file of the networking device.

Cisco IOS CLI Session Usernames

After you have protected access to user EXEC mode and privileged EXEC mode by configuring passwords for them you can further increase the level of security on your networking device by configuring usernames to limit access to CLI sessions to your networking device to specific users.

Usernames that are intended to be used for managing a networking device can be modified with additional options such as:

See the *Cisco IOS Security Command Reference* for more information on how to configure the **username** command.

Cisco IOS Privilege Levels

The default configuration for Cisco IOS software-based networking devices uses privilege level 1 for user EXEC mode and privilege level 15 for privileged EXEC. The commands that can be run in user EXEC mode at privilege level 1 are a subset of the commands that can be run in privileged EXEC mode at privilege 15.

The **privilege** command is used to move commands from one privilege level to another. For example, some ISPs allow their first level technical support staff to enable and disable interfaces to activate new customer connections or to restart a connection that has stopped transmitting traffic. See the Example: Configuring a Device to Allow Users to Shut Down and Enable Interfaces, on page 44 for an example of how to configure this option.

The **privilege** command can also be used to assign a privilege level to a username so that when a user logs in with the username, the session runs at the privilege level specified by the **privilege** command. For example, if you want your technical support staff to view the configuration on a networking device which will help them to troubleshoot network problems without being able to modify the configuration, you can create a username, configure it with privilege level 15, and configure it to run the **show running-config** command automatically. When a user logs in with the username the running configuration will be displayed automatically. The user's session will be logged out automatically after the user has viewed the last line of the configuration. See the Example: Configuring a Device to Allow Users to View the Running Configuration, on page 43 for an example of how to configure this option. To access the running configuration of a device using the **show running-config** command at a privilege level lower than level 15, see the *Configuring a Device to Allow Users to View the Running Configuration* task under the *Protecting Access to Privileged EXEC Mode* section.

These command privileges can also be implemented when you are using AAA with TACACS+ and RADIUS. For example, TACACS+ provides two ways to control the authorization of router commands on a per-user or per-group basis. The first way is to assign privilege levels to commands and have the router verify with the TACACS+ server whether the user is authorized at the specified privilege level. The second way is to explicitly specify in the TACACS+ server, on a per-user or per-group basis, the commands that are allowed. For more information about implementing AAA with TACACS+ and RADIUS, see the technical note How to Assign Privilege Levels with TACACS+ and RADIUS .

Cisco IOS Password Configuration

Cisco IOS software does not prompt you to repeat any passwords that you configure to verify that you have entered the passwords exactly as you intended. New passwords, and changes to existing passwords, go into effect immediately after you press the Enter key at the end of a password configuration command string. If you make a mistake when you enter a new password and have saved the configuration on the networking device to its startup configuration file and exited privileged EXEC mode before you realize that you made a mistake, you may find that you are no longer able to manage the device.

The following are common situations that can happen:

- You make a mistake when configuring a password for local CLI sessions on the console port.
    - If you have properly configured access to your networking device for remote CLI sessions, you can Telnet to it and reconfigure the password on the console port.
- You make a mistake when configuring a password for remote Telnet or SSH sessions.
    - If you have properly configured access to your networking device for local CLI sessions, you can connect a terminal to it and reconfigure the password for the remote CLI sessions.
- You make a mistake when configuring a password for privileged EXEC mode (enable password or enable secret password).
    - You will have to perform a lost password recovery procedure.

- You make a mistake when configuring your username password, and the networking device requires that you log in to it with your username.
    - If you do not have access to another account name, you will have to perform a lost password recovery procedure.

To protect yourself from having to perform a lost password recovery procedure open two CLI sessions to the networking device and keep one of them in privileged EXEC mode while you reset the passwords using the other session. You can use the same device (PC or terminal) to run the two CLI sessions or two different devices. You can use a local CLI session and a remote CLI session or two remote CLI sessions for this procedure. The CLI session that you use to configure the password can also be used to verify that the password was changed properly. The other CLI session that you keep in privileged EXEC mode can be used to change the password again if you made a mistake the first time you configured it.

You should not save password changes that you have made in the running configuration to the startup configuration until you have verified that your password was changed successfully. If you discover that you made a mistake configuring a password, and you were not able to correct the problem using the local and remote CLI session technique, you can power cycle the networking device so that it returns to the previous passwords that are stored in the startup configuration.

Product Security Baseline Password Encryption and Complexity Restrictions

Product security baseline (PSB) mandates basic security functions and features for all Cisco platforms and products.

There are 12 priority security requirements out of the 110 mandatory requirements in version 2.0 of the product security baseline that must be met to allow the shipping of any product.

The following two sections discuss restrictions that are relevant in AAA technology:

- Password complexity restrictions
- Protection of stored credentials

Password Complexity Restrictions

The PSB states the following requirements for password complexity restrictions on Cisco products:

- Whenever a user or an administrator wants to create or change a password, the following restrictions apply to the products:
    - The new password contains characters from at least three of the following classes: lowercase letters, uppercase letters, digits, and special characters.
    - No character in the new password should be repeated more than three times consecutively.
    - The new password should not be the same as the associated username, and should not be the username reversed. The password obtained by capitalization of the username or username reversed also is not accepted.
    - The new password should not be “cisco”, “ocsic”, or any variant obtained by changing the capitalization of letters therein, or by substituting “1”, “|”, or “!” for i, and substituting “0” for “o”, and substituting “$” for “s”.

- It must be possible to individually enable or disable each of these restrictions as part of the product configuration. A user interface should be available for one time to override the restrictions when a password is being set by an administrator.

The first restriction need not be applied to passwords that are expected to be used via a numerical pin pad; in this case, passwords consisting only of digits are permitted. However, such passwords must be used only for access to messaging services, and not for general computer networking services.

For an administrator to enable the restrictions, no particular default setting is required. Restrictions should be enabled by default on products that permit nonadministrative end users to change their own passwords.

AAA enforces these restrictions on creating passwords used in a AAA context which includes passwords created using the **username** command and passwords created to download authorization data.

The complexity restrictions are enabled or disabled using the **aaa password restriction** command. The behavior should be backward-compatible in allowing passwords that were configured before the complexity restrictions were enabled. The CLI should be disabled by default. When the CLI is enabled on a running device, the passwords configured prior to enabling the command should not be subject to the complexity restrictions. The passwords configured following the command should be subject to complexity restrictions. When a device is rebooted using a startup configuration containing the password complexity command enabled, the passwords present in the startup configuration should be allowed without the complexity restrictions; any passwords that are configured after the device has booted should be subject to the complexity restrictions.

Protection of Stored Credentials

The PSB states the following requirement for password complexity restrictions on Cisco products:

- If the product authenticates remote entities using protocols that do not require the product to possess recoverable copies of the remote entities' credentials, then no recoverable copies of credentials which are used only in this way are to be stored.
- In the specific case where the product authenticates remote entities using the traditional password interchange in which the remote entity discloses its credential to the product for direct comparison against a database, stored credentials must be protected by a method at least as strong as a SHA-1 digest. The use of SHA-256 or SHA-384 instead of SHA-1 is recommended.

To be compliant with the PSB, AAA enforces the protection of stored credentials using SHA-256.

Blocking Repeat Failed Login Attempts

Before You Begin

If the user attempts to login multiple times within a set time period, the device blocks further login attempts for a preset time duration. For example, if three unsuccessful login attempts are made within, 20 seconds, the user is blocked for the next 300 seconds.

Sample Configuration

```
Device(config)#aaa authentication rejected ?
  <1-65535>  Fail attempts max value
Device(config)#aaa authentication rejected 3 ?
  in  Watch time period

Device(config)#aaa authentication rejected 3 in 20 ?
  ban  Ban time period
```

```
Device(config)#aaa authentication rejected 3 in 20 ban 300
```

Recovering from a Lost or Misconfigured Password for Local Sessions

Three methods can be used to recover a lost or misconfigured password for local CLI sessions over console port. The method that you will use depends on the current configuration of your networking device.

The following sections describes the three methods that can be used to recover a lost or misconfigured password:

Networking Device Is Configured to Allow Remote CLI Sessions

The fastest method to recover from a lost or misconfigured password for local CLI sessions is to establish a remote CLI session with the networking device and repeat the steps in the Configuring a Password for Local CLI Sessions, on page 24. Your networking device must be configured to allow remote CLI sessions and you must know the remote CLI session password to perform this procedure.

Networking Device Is Not Configured to Allow Remote CLI Sessions

- If you cannot establish a remote session to your networking device, and you have not saved the misconfigured local CLI session password to the startup configuration, you can restart the networking device. When the networking device starts up again it will read the startup configuration file. The previous local CLI session password is restored.

Caution Restarting a networking device will cause it to stop forwarding traffic. This will also cause an interruption in any services that are running on the networking device, such as a DHCP server service. You should restart a networking device only during a period of time that has been allocated for network maintenance.

Recovering from a Lost or Misconfigured Password for Remote Sessions

Three methods that can be used to recover from a lost or misconfigured remote CLI session password. The method that you will use depends on the current configuration of your networking device.

Networking Device Is Configured to Allow Local CLI Sessions

The fastest method to recover from a lost or misconfigured password for remote CLI sessions is to establish a local CLI session with the networking device and repeat the steps in the Configuring a Password for Remote CLI Sessions, on page 22. Your networking device must be configured to allow local CLI sessions and you must know the local CLI session password to perform this procedure.

Networking Device Is Not Configured to Allow Local CLI Sessions

- If you cannot establish a local CLI session to your networking device, and you have not saved the misconfigured remote CLI session password to the startup configuration, you can restart the networking

device. When the networking device starts up again it will read the startup configuration file. The previous remote CLI session password is restored.

Caution Restarting a networking device will cause it to stop forwarding traffic. This will also cause an interruption in any services that are running on the networking device, such as a DHCP server service. You should restart a networking device only during a period of time that has been allocated for network maintenance.

Recovering from Lost or Misconfigured Passwords for Privileged EXEC Mode

Two methods can be used to recover from a lost or misconfigured privileged EXEC mode password. The method that you will use depends on the current configuration of your networking device.

A Misconfigured Privileged EXEC Mode Password Has Not Been Saved

- If you have not saved the misconfigured privileged EXEC mode password to the startup configuration, you can restart the networking device. When the networking device starts up again it will read the startup configuration file. The previous privileged EXEC mode password is restored.

Caution Restarting a networking device will cause it to stop forwarding traffic. This will also cause an interruption in any services that are running on the networking device, such as a DHCP server service. You should restart a networking only device during a period of time that has been allocated for network maintenance.

How to Configure Security with Passwords, Privileges, and Logins

Protecting Access to User EXEC Mode

Configuring a Password for Remote CLI Sessions

This task will assign a password for remote CLI sessions. After you have completed this task the networking device will prompt you for a password the next time that you start a remote CLI session with it.

Cisco IOS software-based networking devices require that you have a password configured for remote CLI sessions. If you attempt to start a remote CLI session with a device that does not have a password configured for remote CLI sessions you will see a message that a password is required and the password is not set. The remote CLI session will be terminated by the remote host.

Before You Begin

If you have not previously configured a password for remote CLI sessions, you must perform this task over a local CLI session using a terminal or a PC running a terminal emulation application attached to the console port.

Your terminal, or terminal emulation application, must be configured with the settings that are used by the console port on the networking device. The console ports on most Cisco networking devices require the following settings: 9600 baud, 8 data bits, 1 stop bit, no parity, and flow control is set to “none”. See the documentation for your networking device if these settings do not work for your terminal.

Note If you have not previously configured a password for remote CLI sessions, you must perform this task over a local CLI session using a terminal attached to the console port.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **line vty** *line-number* [*ending-line-number*]
4. **password** *password*
5. **end**
6. **telnet** *ip-address*
7. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**line vty** *line-number* [*ending-line-number*] **Example:** `Device(config)# line vty 0 4`	Enters line configuration mode.
Step 4	**password** *password*	Assigns a password for remote CLI session.

	Command or Action	Purpose
	Example: `Device(config-line)# password H7x3U8`	• The argument *password* is a character string that specifies the line password. The following rules apply to the *password* argument: • The first character cannot be a number. • The string can contain any alphanumeric characters, including spaces, up to 80 characters. You cannot specify the password in the format number-space-anything. • Passwords are case-sensitive.
Step 5	**end** **Example:** `Device(config-line)# end`	Exits the current configuration mode and returns to privileged EXEC mode.
Step 6	**telnet** *ip-address* **Example:** `Device# telnet 172.16.1.1`	Start a remote CLI session with the networking device from your current CLI session using the IP address of an interface in the networking device that is in an operational state (interface up, line protocol up). • Enter the password that you configured is Step 4 when prompted. • To perform this step, your networking device must have an interface that is in an operational state. The interface must have a valid IP address. **Note** This procedure is often referred to as a starting recursive Telnet session because you are initiating a remote Telnet session with the networking device from the networking device itself.
Step 7	**exit** **Example:** `Device# exit`	Terminates the remote CLI session (recursive Telnet session) with the networking device.

Troubleshooting Tips

Repeat this task if you made a mistake when configuring the remote CLI session password.

Configuring a Password for Local CLI Sessions

This task will assign a password for local CLI sessions over the console port. After you have completed this task, the networking device will prompt you for a password the next time that you start a local CLI session on the console port.

This task can be performed over a local CLI session using the console port or a remote CLI session. If you want to perform the optional step of verifying that you have configured the password correctly you should perform this task using a local CLI session using the console port.

Before You Begin

If you want to perform the optional step of verifying the local CLI session password, you must perform this task using a local CLI session. You must have a terminal or a PC running a terminal emulation program connected to the console port of the networking device. Your terminal must be configured with the settings that are used by the console port on the networking device. The console ports on most Cisco networking devices require the following settings: 9600 baud, 8 data bits, 1 stop bit, no parity, and flow control set to "none". See the documentation for your networking device if these settings do not work for your terminal.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **line console** *line-number*
4. **password** *password*
5. **end**
6. **exit**
7. Press the Enter key, and enter the password from Step 4 when prompted.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**line console** *line-number* **Example:** `Device(config)# line console 0`	Enters line configuration mode and selects the console port as the line that you are configuring.
Step 4	**password** *password* **Example:** `Device(config-line)# password Ji8F5Z`	Assigns a password for local CLI session over the console port. • The argument *password* is a character string that specifies the line password. The following rules apply to the *password* argument: • The first character cannot be a number.

	Command or Action	Purpose
		• The string can contain any alphanumeric characters, including spaces, up to 80 characters. You cannot specify the password in the format number-space-anything. • Passwords are case-sensitive.
Step 5	**end** **Example:** `Device(config-line)# end`	Exits the current configuration mode and returns to privileged EXEC mode.
Step 6	**exit** **Example:** `Device# exit`	Exits privileged EXEC mode.
Step 7	Press the Enter key, and enter the password from Step 4 when prompted.	(Optional) Initiates the local CLI session on the console port. • Enter the password that you configured is Step 4 when prompted to verify that it was configured correctly. **Note** This step can be performed only if you are using a local CLI session to perform this task.

Troubleshooting Tips

If your new password is not accepted proceed to the Configuration Examples for Configuring Security with Passwords Privileges and Logins for instructions on what to do next.

Protecting Access to Privileged EXEC Mode

Configuring the Enable Password

Cisco no longer recommends that you use the **enable password** command to configure a password for privileged EXEC mode. The password that you enter with the **enable password**command is stored as plain text in the configuration file of the networking device. You can encrypt the password for the **enable password**command in the configuration file of the networking device using the **service password-encryption** command. However the encryption level used by the **service password-encryption** command can be decrypted using tools available on the Internet.

Instead of using the **enable password**command, Cisco recommends using the **enable secret** commandbecause it encrypts the password that you configure with strong encryption. For more information on password

encryption issues see the Cisco IOS Password Encryption Levels, on page 16. For information on configuring the **enable secret**command see the Configuring the Enable Secret Password, on page 29.

Note The networking device must not have a password configured by the **enable secret** command in order for you to perform this task successfully. If you have already configured a password for privileged EXEC mode using the **enable secret**command, that the password configured takes precedences over the password that you configure in this task using the **enable password** command.

You cannot use the same password for the **enable secret**command and the **enable** *password* command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **enable password** *password*
4. **end**
5. **exit**
6. **enable**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**enable password** *password* **Example:** `Device(config)# enable password t6D77CdKq`	Configures a password for privileged EXEC mode. • The argument *password* is a character string that specifies the enable password. • The argument *password*must contain from 1 to 25 uppercase and lowercase alphanumeric characters. • The argument *password*must not have a number as the first character. • The argument *password*can have leading spaces, but they are ignored. However, intermediate and trailing spaces are recognized. • The argument *password*can contain the question mark (?) character if you precede the question mark with the key combination Crtl-v when

	Command or Action	Purpose
		you create the password; for example, to create the password abc?123, do the following: • Enter abc • Type Crtl-v • Enter ?123
Step 4	**end** **Example:** `Device(config)# end`	Exits the current configuration mode and returns to privileged EXEC mode.
Step 5	**exit** **Example:** `Device# exit`	Exits privileged EXEC mode and enters user EXEC mode.
Step 6	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter the password you configured in Step 3.

Troubleshooting Tips

If your new password is not accepted, proceed to the Recovering from Lost or Misconfigured Passwords for Privileged EXEC Mode section for instructions on what to do next.

Configuring Password Encryption for Clear Text Passwords

Cisco IOS software stores passwords in clear text in network device configuration files for several features such as passwords for local and remote CLI sessions, and passwords for neighbor authentication for routing protocols. Clear text passwords are a security risk because anybody with access to archived copies of the configuration files can discover the passwords that are stored as clear text. The **service password-encryption**command can be used to encrypt clear text commands in the configuration files of networking devices. See the Cisco IOS Password Encryption Levels, on page 16 for more information.

Complete the following steps to configure password encryption for passwords that are stored as clear text in the configuration files of your networking device.

Before You Begin

You must have at least one feature that uses clear text passwords configured on your networking device for the **service password-encryption**command to have any immediate effect.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **service password-encryption**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**service password-encryption** **Example:** `Device(config)# service password-encryption`	Configures password encryption for all passwords, clear text passwords, including username passwords, authentication key passwords, the privileged command password, console and virtual terminal line access passwords, and BGP neighbor passwords.
Step 4	**end** **Example:** `Device(config)# end`	Exits the current configuration mode and returns to privileged EXEC mode.

Configuring the Enable Secret Password

Cisco recommends that you use the **enable secret** command, instead of the **enable** *password* command to configure a password for privileged EXEC mode. The password created by the **enable secret** command is encrypted with the more secure MD5 algorithm.

Note You cannot use the same password for the **enable secret** command and the **enable** *password* command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. Do one of the following:
   - **enable secret** *unencrypted-password*
   - **enable secret** *encryption-type encrypted-password*
4. **end**
5. **exit**
6. **enable**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

Step 3	Do one of the following:
	• **enable secret** *unencrypted-password*
	• **enable secret** *encryption-type encrypted-password*
	
	Example:
	`Device(config)# enable secret t6D77CdKq`
	
	Example:
	`Device(config)# enable secret 5 $1$/x6H$RhnDI3yLC4GA01aJnHLQ4/`
		Configures a password for privileged EXEC mode.
		• The argument *password* is a character string that specifies the **enable secret** password. The following rules apply to the *password* argument:
		• The argument *password* must contain from 1 to 25 uppercase and lowercase alphanumeric characters.
		• The argument *password* must not have a number as the first character.
		• The argument *password* can have leading spaces, but they are ignored. However, intermediate and trailing spaces are recognized.
		• The argument *password* can contain the question mark (?) character if you precede the question mark with the key combination Crtl-v when you create the password; for example, to create the password abc?123, do the following:
		  • Enter abc
		  • Type Crtl-v
		  • Enter ?123

	Command or Action	Purpose
		or Sets a previously encrypted password for privileged EXEC mode by entering the number 5 before the previously encrypted string. • You must enter an exact copy of a password from a configuration file that was previously encrypted by the **enable secret** command to use this method.
Step 4	**end** **Example:** `Device(config)# end`	Exits the current configuration mode and returns to privileged EXEC mode.
Step 5	**exit** **Example:** `Device# exit`	Exits privileged EXEC mode and enters user EXEC mode.
Step 6	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter the password that you configured in Step 3.

Troubleshooting Tips

If your new password is not accepted proceed to the Recovering from Lost or Misconfigured Passwords for Privileged EXEC Mode, on page 22 for instructions on what to do next.

Configuring a Device to Allow Users to View the Running Configuration

To access the running configuration of a device using the **show running-config** command at a privilege level lower than level 15, perform the following task.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **privilege exec all level** *level command-string*
4. **file privilege** *level*
5. **privilege configure all level** *level command-string*
6. **end**
7. **show privilege**
8. **show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**privilege exec all level** *level command-string* **Example:** `Device(config)# privilege exec all level 5 show running-config`	Changes the privilege level of the specified command from one privilege level to another.
Step 4	**file privilege** *level* **Example:** `Device(config)# file privilege 5`	Allows a user of the privilege level to execute commands that involve the file system on a device.
Step 5	**privilege configure all level** *level command-string* **Example:** `Device(config)# privilege configure all level 5 logging`	Allows a user of a privilege level to see specific configuration commands. For example, allows the user of privilege level 5 to see the logging configuration commands in the running configuration.
Step 6	**end** **Example:** `Device(config)# end`	Exits global configuration mode and returns to privileged EXEC mode.

	Command or Action	Purpose
Step 7	**show privilege** **Example:** `Device# show privilege`	Displays the current privilege level.
Step 8	**show running-config** **Example:** `Device# show running-config`	Displays the current running configuration for the specified privilege level.

The following output for the **show running-config** command displays the logging configuration commands in the running configuration. Users with a privilege level below 15 can view the running configuration after configuring the **privilege configure all level** *level command-string* command.

```
Device# show running-config

Building configuration...

Current configuration : 128 bytes
!
boot-start-marker
boot-end-marker
!
no logging queue-limit
logging buffered 10000000
no logging rate-limit
!
!
!
end
```

Configuring Security Options to Manage Access to CLI Sessions and Commands

The tasks in this section describe how to configure your networking device to permit the use of a subset of privileged EXEC mode commands by users who should not have access to all of the commands that are available in privileged EXEC mode.

These tasks are beneficial for companies that have multiple levels of network support staff and the company wants the staff at each level to have access to a different subset of the privileged EXEC mode commands.

In this task the users who should not have access to all of the commands that are available in privileged EXEC mode are referred to as the first-line technical support staff.

This section contains the following procedures:

Configuring the Networking Device for the First-Line Technical Support Staff

This task describes how to configure the networking device for first-line technical support users. First-line technical support staff are usually not allowed to run all of the commands available in privileged EXEC mode

(privilege level 15) on a networking device. They are prevented from running commands that they are not authorized for by not being granted access to the password assigned to privileged EXEC mode or to other roles that have been configured on the networking device.

The **privilege** command is used to move commands from one privilege level to another in order to create the additional levels of administration of a networking device that is required by companies that have different levels of network support staff with different skill levels.

The default configuration of a Cisco IOS device permits two types of users to access the CLI. The first type of user is allowed to access only user EXEC mode. The second type of user is allowed access to privileged EXEC mode. A user who is allowed to access only user EXEC mode is not allowed to view or change the configuration of the networking device, or to make any changes to the operational status of the networking device. However, a user who is allowed access to privileged EXEC mode can make any change to a networking device that is allowed by the CLI.

In this task the two commands that normally run at privilege level 15 are reset to privilege level 7 using the **privilege** command so that first-line technical support users will be allowed to run the two commands. The two commands for which the privilege levels will be reset are the **clear counters** command and **reload** command.

- The **clear counters** command is used to reset the counter fields on interfaces for statistics such as packets received, packets transmitted, and errors. When a first-line technical support user is troubleshooting an interface-related connectivity issue between networking devices, or with remote users connecting to the network, it is useful to reset the interface statistics to zero and then monitor the interfaces for a period of time to see if the values in the interface statistics counters change.
- The **reload** command is used initiate a reboot sequence for the networking device. One common use of the reload command by the first-line technical support staff is to cause the networking device to reboot during a maintenance window so that it loads a new operating system that was previously copied onto the networking device's file system by a user with a higher level of authority.

Any user that is permitted to know the **enable secret** password that is assigned to the first-line technical support user role privilege level can access the networking device as a first-line technical support user. You can add a level of security by configuring a username on the networking device and requiring that the users know the username and the password. Configuring a username as an additional level of security is described in the Configuring a Device to Require a Username for the First-Line Technical Support Staff, on page 39.

Before Cisco IOS Releases 12.0(22)S and 12.2(13)T, each command in a privilege level had to be specified with a separate **privilege** command. In Cisco IOS Releases 12.0(22)S, 12.2(13)T, and later releases, a "wildcard" option specified by the new keyword **all** was introduced that allows you to configure access to multiple commands with only one **privilege** command. By using the new **all** keyword, you can specify a privilege level for all commands which begin with the string you enter. In other words, the **all** keyword allows you to grant access to all command-line options and suboptions for a specified command.

For example, if you wanted to create a privilege level to allow users to configure all commands which begin with **service-module t1** (such as **service-module t1 linecode** or **service-module t1 clock source**) you can use the **privilege interface all level 2 service-module t1** command instead of having to specify each **service-module t1** command separately.

If the command specified in the **privilege** command (used with the **all** keyword) enables a configuration submode, all commands in the submode of that command will also be set to the specified privilege level.

Note The **all** "wildcard" keyword option for the **privilege** command is not supported in versions of Cisco IOS software prior to Cisco IOS Releases 12.0(22)S and, 12.2(13)T.

You must not have the **aaa new-model** command enabled on the networking device. You must not have the **login local** command configured for the local CLI sessions over the console port or the remote CLI sessions.

Note For clarity, only the arguments and keywords that are relevant for each step are shown in the syn the steps in this task. See the Cisco IOS command reference book for your Cisco IOS release for information on the additional arguments and keywords that can be used with these commands.

Caution Do not use the **no** form of the **privilege** command to reset the privilege level of a command to its because it might not return the configuration to the correct default state. Use the **reset** keyword fo **privilege** command instead to return a command to its default privilege level. For example, to rem **privilege exec level** *reload* command from the configuration and return the **reload** command to its privilege of 15, use the **privilege exec reset** *reload* command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **enable secret level** *level password*
4. **privilege exec level** *level command-string*
5. **privilege exec all level** *level command-string*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**enable secret level** *level password* **Example:** `Device(config)# enable secret level 7 Zy72sKj`	Configures a new enable secret password for privileged EXEC mode.
Step 4	**privilege exec level** *level command-string* **Example:** `Device(config)# privilege exec level 7 clear counters`	Changes the privilege level of the **clear counters** command from one privilege level to another.
Step 5	**privilege exec all level** *level command-string* **Example:** `Device(config)# privilege exec all level 7 reload`	Changes the privilege level of the **reload** command from one privilege level to another.
Step 6	**end** **Example:** `Device(config)# end`	Exits global configuration mode.

Verifying the Configuration for the First-Line Technical Support Staff

This task describes how to verify that the network device is configured correctly for the first-line technical support staff.

Before You Begin

The following commands must have been modified to run at privilege level 7 for this task:

- **clear counters**
- **reload**

SUMMARY STEPS

1. **enable** *level password*
2. **show privilege**
3. **clear counters**
4. **clear ip route ***
5. **reload in** *time*
6. **reload cancel**
7. **disable**
8. **show privilege**

DETAILED STEPS

Step 1 **enable** *level password*
Logs the user in into the networking device at the privilege level specified for the *level* argument.

Example:

```
Device> enable 7 Zy72sKj
```

Step 2 **show privilege**
Displays the privilege level of the current CLI session.

Example:

```
Device# show privilege

Current privilege level is 7
```

Step 3 **clear counters**
Clears the interface counters.

Example:

```
Device# clear counters

Clear "show interface" counters on all interfaces [confirm]
Device#
02:41:37: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
```

Step 4 **clear ip route ***
The **clear ip route**command should not be allowed because it was never changed from the default privilege 15 to the privilege level 7.

Example:

```
Device# clear ip route *
                       ^
```

```
% Invalid input detected at '^' marker.
```

Step 5 **reload in** *time*

Causes the networking device to reboot.

Example:

```
Device# reload in 10

Reload scheduled in 10 minutes by console
Proceed with reload? [confirm]
Router#
***
*** --- SHUTDOWN in 0:10:00 ---
***
02:59:50: %SYS-5-SCHEDULED_RELOAD: Reload requested for 23:08:30 PST Sun Mar 20
```

Step 6 **reload cancel**

The **reload cancel** command terminates a reload that was previously set up with the **reload in** *time* command.

Example:

```
Device# reload cancel

***
*** --- SHUTDOWN ABORTED ---
***
04:34:08: %SYS-5-SCHEDULED_RELOAD_CANCELLED:  Scheduled reload cancelled at 15:38:46 PST Sun Mar 27
 2005
```

Step 7 **disable**

Exits the current privilege level and returns to privilege level 1.

Example:

```
Device# disable
```

Step 8 **show privilege**

Displays the privilege level of the current CLI session.

Example:

```
Device> show privilege

Current privilege level is 1
```

Troubleshooting Tips

If your configuration does not work the way that you want it to and you want to remove the privilege commands from the configuration, use the **reset** keyword for the **privilege** command to return the commands to their

default privilege level. For example, to remove the command **privilege exec level** *reload* command from the configuration and return the **reload** command to its default privilege of 15 use the **privilege exec reset** *reload*command.

Configuring a Device to Require a Username for the First-Line Technical Support Staff

This task configures the networking device to require that the first-line technical support staff log in to the networking device with a login name of admin. The admin username configured in this task is assigned the privilege level of 7, which will allow users who log in with this name to run the commands that were reassigned to privilege level 7 in the Configuring the Networking Device for the First-Line Technical Support Staff, on page 33 task. When a user successfully logs in with the admin username, the CLI session will automatically enter privilege level 7.

Before Cisco IOS Releases 12.0(18)S and 12.2(8)T, two types of passwords were associated with usernames: Type 0, which is a clear text password visible to any user who has access to privileged mode on the router, and type 7, which has a password encrypted by the **service password encryption** command.

In Cisco IOS Releases 12.0(18)S, 12.2(8)T, and later releases, the new **secret** keyword for the **username** command allows you to configure MD5 encryption for username passwords.

Before You Begin

The following commands must have been modified to run at privilege level 7 for this task:

- **clear counters**
- **reload**

See the Configuring the Networking Device for the First-Line Technical Support Staff, on page 33 for instructions on how to change the privilege level for a command.

Note MD5 encryption for the **username** command is not supported in versions of Cisco IOS software prior to Cisco IOS Releases 12.0(18)S and 12.2(8)T.

You must not have the **aaa-new model** command enabled on the networking device. You must not have the **login local** command configured for the local CLI sessions over the console port or the remote CLI sessions.

Note For clarity, only the arguments and keywords that are relevant for each step are shown in the syn the steps in this task. Refer to the Cisco IOS command reference book for your Cisco IOS releas further information on the additional arguments and keywords that can be used with these comm

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **username** *username* **privilege** *level* **secret** *password*
4. **end**
5. **disable**
6. **login**
7. **show privilege**
8. **clear counters**
9. **clear ip route ***
10. **reload in** *time*
11. **reload cancel**
12. **disable**
13. **show privilege**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enters privileged EXEC mode. Enter the password when prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**username** *username* **privilege** *level* **secret** *password* **Example:** `Device(config)# username admin privilege 7 secret Kd65xZa`	Creates a username and applies MD5 encryption to the *password* text string.
Step 4	**end** **Example:** `Device(config)# end`	Exits global configuration mode.

	Command or Action	Purpose
Step 5	**disable** **Example:** `Device# disable`	Exits the current privilege level and returns to user EXEC mode.
Step 6	**login** **Example:** `Device# login`	Logs in the user. Enter the username and password you configured in Step 3 when prompted.

Step 7	**show privilege**
	
	Example:
	`Device# show privilege`
	
	`Current privilege level is 7`
		The **show privilege** command displays the privilege level of the CLI session.

Step 8	**clear counters**
	
	Example:
	`Device# clear counters`
	
	`Clear "show interface" counters on all interfaces [confirm]`
	`Device#`
	`02:41:37: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console`
		The **clear counters** command clears the interface counters.

Step 9	**clear ip route ***
	
	Example:
	`Device# clear ip route *`
	`                  ^`
	`% Invalid input detected at '^' marker.`
		The **clear ip route**command is not allowed because it was never changed from the default privilege 15 to the privilege level 7.

Step 10	**reload in** *time*
	
	Example:
	`Device# reload in 10`
	
	`Reload scheduled in 10 minutes by console`
	`Proceed with reload? [confirm]`
	`Router#`
	`***`
	`*** --- SHUTDOWN in 0:10:00 ---`
	`***`
	`02:59:50: %SYS-5-SCHEDULED_RELOAD: Reload requested for 23:08:30 PST Sun Mar 20`
		The reload command causes the networking device to reboot.

	Command or Action	Purpose

Step 11	**reload cancel**
	
	Example:
	`Device# reload cancel`
	`*** *** --- SHUTDOWN ABORTED --- *** 04:34:08: %SYS-5-SCHEDULED_RELOAD_CANCELLED: Scheduled reload cancelled at 15:38:46 PST Sun Mar 27 2005`
		Terminates a reload that was previously set up with the **reload in** *time* command.

Step 12	**disable** **Example:** `Device# disable`	Exits the current privilege level and returns to user EXEC mode.

Step 13	**show privilege**
	
	Example:
	`Device> show privilege`
	`Current privilege level is 1`
		Displays the privilege level of the current CLI session.

Configuration Examples for Configuring Security with Passwords, Privileges, and Logins

Example: Configuring a Device to Allow Users to Clear Remote Sessions

The following example shows how to configure a networking device to allow a nonadministrative user to clear remote CLI session vty lines.

The first section is an excerpt of the running configuration for this example. The following sections show you how this example is used.

The following section is an excerpt of the running configuration:

```
!
privilege exec level 7 clear line
!
no aaa new-model
!
!
username admin privilege 7 secret 5 $1$tmIw$1aM7sadKhWMpkVTzxNw1J.
!
privilege exec level 7 clear line
!
```

```
! the privilege exec level 7 clear command below is entered automatically
! when you enter the privilege exec level 7 clear line command above, do
! not enter it again
!
privilege exec level 7 clear
!
```

The following section using the **login** command shows the user logging in to the networking device with the username of admin:

```
Device> login
Username: admin
Password:
```

The following section using the **show privilege** command shows that the current privilege level is 7:

```
Device# show privilege

Current privilege level is 7
```

The following section using the **show user** command shows that two users (admin and root) are currently logged in to the networking device:

```
Device# show user

    Line       User       Host(s)              Idle       Location
*  0 con 0     admin      idle                 00:00:00
   2 vty 0     root       idle                 00:00:17 172.16.6.2
  Interface    User       Mode                 Idle     Peer Address
```

The following section using the **clear line** command terminates the remote CLI session in use by the username root:

```
Device# clear line 2

[confirm]
 [OK]
```

The following section using the **show user** command shows that admin is the only user logged in to the networking device:

```
Device# show user
    Line       User       Host(s)              Idle       Location
*  0 con 0     admin      idle                 00:00:00
  Interface    User       Mode                 Idle     Peer Address
```

Example: Configuring a Device to Allow Users to View the Running Configuration

For Users With Privilege Level 15

The following example shows how to configure a networking device to allow a nonadministrative users (no access to privileged EXEC mode) to view the running configuration automatically. This example requires that the username is configured for privilege level 15 because many of the commands in the configuration file can be viewed only by users who have access to privilege level 15.

The solution is to temporarily allow the user access to privilege level 15 while running the **show running-config** command and then terminating the CLI session when the configuration file has been viewed. In this example the networking device will automatically terminate the CLI session when the end of the configuration file has been viewed. No further configuration steps are required.

Caution You must include the **noescape** keyword for the **username** command to prevent the user from entering an escape character that will terminate viewing the configuration file and leave the session running at privilege level 15.

```
!
!
username viewconf privilege 15 noescape secret 5 $1$zA9C$TDWD/Q0zwp/5xRwRqdgc/.
username viewconf autocommand show running-config
!
```

For Users With Privilege Level Lower Than Level 15

The following example shows how to configure a networking device to allow a user with privilege level lower than level 15 to view the running configuration.

```
Device> enable
Device# configure terminal
Device(config)# privilege exec all level 5 show running-config
Device(config)# file privilege 5
Device(config)# privilege configure all level 5 logging
Device(config)# end
Device# show privilege

Current privilege level is 5

Device# show running-config

Building configuration...

Current configuration : 128 bytes
!
boot-start-marker
boot-end-marker
!
no logging queue-limit
logging buffered 10000000
no logging rate-limit
!
!
!
end
```

Example: Configuring a Device to Allow Users to Shut Down and Enable Interfaces

The following example shows how to configure a networking device to allow nonadministrative users to shut down and enable interfaces.

The first section is an excerpt of the running configuration for this example. The following sections show you how this example is used.

The following section is an excerpt of the running configuration:

```
!
no aaa new-model
!
username admin privilege 7 secret 5 $1$tmIw$1aM7sadKhWMpkVTzxNw1J.
!
privilege interface all level 7 shutdown
privilege interface all level 7 no shutdown
privilege configure level 7 interface
privilege exec level 7 configure terminal
!
! the privilege exec level 7 configure command below is entered automatically
! when you enter the privilege exec level 7 configure terminal command above, do
! not enter it again
!
privilege exec level 7 configure
!
```

The following section using the **login** command shows the user logging in to the networking device with the username of admin:

```
Device> login
Username: admin
Password:
```

The following section using the **show privilege** command shows that the current privilege level is 7:

```
Device# show privilege

Current privilege level is 7
```

The following section using the **show user** command shows that admin is the only user logged in to the networking device:

```
Device# show user

    Line       User       Host(s)              Idle       Location
*  0 con 0     admin      idle                 00:00:00
  Interface    User       Mode                 Idle     Peer Address
```

The following section shows that the admin user is permitted to shut down and enable an interface:

```
Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# interface ethernet 1/0
Device(config-if)# shutdown
Device(config-if)# no shutdown
Device(config-if)# exit
Device#
```

Where to Go Next

Once you have established a baseline of security for your networking devices you can consider more advanced options such as:

- Role-based CLI access—The role-based CLI access feature offers a more comprehensive set of options than the **privilege** command (described in this document) for network managers who want to allow different levels of technical support staff to have different levels of access to CLI commands.

- AAA security—Many Cisco networking devices offer an advanced level of security using AAA features. All of the tasks described in this document, and other—more advanced security features—can be implemented using AAA on the networking device in conjunction with a remote TACACS+ or RADIUS server. For information how to configure AAA security features that can be run locally on a networking device, or for information on how to configure remote AAA security using TACACS+ or RADIUS servers, see the *Securing User Services Configuration Guide Library* .

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
Security commands	*Cisco IOS Security Command Reference*
Managing user access to CLI commands and configuration information	Role-Based CLI Access
Configuring MD5 secure neighbor authentication for protocols such as OSPF and BGP	Neighbor Router Authentication: Overview and Guidelines
Assigning privilege levels with TACACS+ and RADIUS	How to Assign Privilege Levels with TACACS+ and RADIUS

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Configuring Security with Passwords, Privileges, and Logins

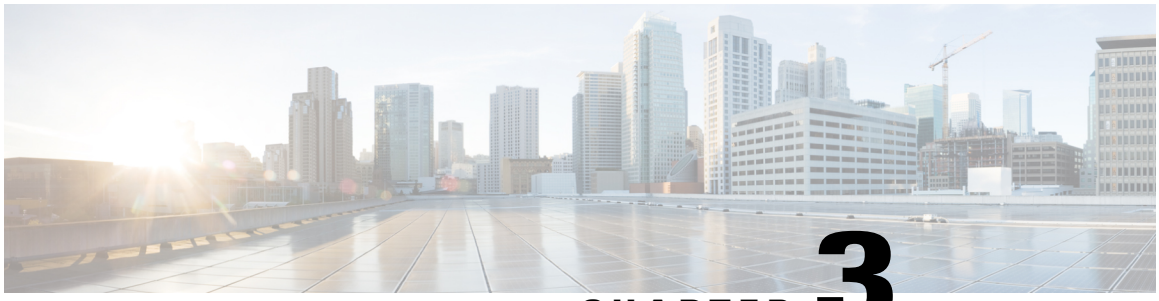
The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 2: Feature Information for Configuring Security with Passwords, Privilege Levels, and Login Usernames for CLI Sessions on Networking Devices

Feature Name	Releases	Feature Configuration Information
Enhanced Password Security	12.0(18)S 12.2(8)T	Using the Enhanced Password Security feature, you can configure MD5 encryption for username passwords. MD5 encryption is a one-way hash function that makes reversal of an encrypted password impossible, providing strong encryption protection. Using MD5 encryption, you cannot retrieve clear text passwords. MD5 encrypted passwords cannot be used with protocols that require that the clear text password be retrievable, such as Challenge Handshake Authentication Protocol (CHAP).
Privilege Command Enhancement	12.0(22)S 12.2(13)T	The **all** keyword was added to the **privilege** command as a wild card to reduce the number of times that the **privilege** command is entered when you are changing the privilege level of several keywords for the same command.

Feature Name	Releases	Feature Configuration Information
Product Security Baseline: Password Encryption and Complexity Restrictions	15.0(1)S 15.1(1)SG 15.1(1)SY	This feature enforces restrictions on creating passwords used in a AAA context that includes passwords created through the **username** command and passwords created to download authorization data. The following commands were introduced or modified: **aaa password restriction**, **enable secret**, **username secret**.



CHAPTER 3

No Service Password-Recovery

The No Service Password-Recovery feature is a security enhancement that prevents anyone with console access from accessing the router configuration and clearing the password. It also prevents anyone from changing the configuration register values and accessing NVRAM.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for No Service Password-Recovery

You must download and install ROM monitor (ROMMON) version 12.2(11)YV1 before you can use this feature.

Information About No Service Password-Recovery

Cisco Password Recovery Procedure

The Cisco IOS software provides a password recovery procedure that relies upon gaining access to ROMMON mode using the Break key during system startup. When the router software is loaded from ROMMON mode, the configuration is updated with the new password.

The password recovery procedure enables anyone with console access, the ability to access the router and its network. The No Service Password-Recovery feature prevents the completion of the Break key sequence and the entering of ROMMON mode during system startups and reloads.

Configuration Registers and System Boot Configuration

The lowest four bits of the configuration register (bits 3, 2, 1, and 0) form the boot field. The boot field determines if the router boots manually from ROM or automatically from flash or the network. For example, when the configuration register boot field value is set to any value from 0x2 to 0xF, the router uses the boot field value to form a default boot filename for autobooting from a network server.

Bit 6, when set, ignores the startup configuration, while bit 8 enables a break. To use the No Security Password Recovery feature, you must set the configuration register to autoboot before it can be enabled. Any other configuration register setting will prevent the feature from being enabled.

Note By default, the no confirm prompt and message are not displayed after reloads.

How to Enable No Service Password-Recovery

Upgrading the ROMMON Version

If your router or access server does not find a valid system image to load, the system will enter ROMMON mode. ROMMON mode can also be accessed by interrupting the boot sequence during startup.

Another method for entering ROMMON mode is to set the configuration register so that the router automatically enters ROMMON mode when it boots. For information about setting the configuration register value, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide* and *Cisco IOS Network Management Configuration Guide* .

Perform this task to upgrade your version of ROMMON.

SUMMARY STEPS

1. **reload**
2. **set** *tftp-file ip-address ip-subnet-mask default-gateway tftp-server*
3. **sync**
4. **tftpdnld -u**
5. **boot**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**reload** **Example:** `Router> reload`	Reloads a Cisco IOS image. After you issue this command and respond to the system prompts as necessary, the system will begin reloading the system software image. • While the system is reloading, press the Break key or a Break key-combination just after the "Compiled <date> by" message appears. Pressing the Break key interrupts the boot sequence and puts the router into ROMMON mode. **Note** The default Break key combination is Ctrl-C, but this may be configured differently on your system.
Step 2	**set** *tftp-file ip-address ip-subnet-mask default-gateway tftp-server* **Example:** `ROMMON> set tftpabc 10.10.0.0 255.0.0.0 10.1.1.0 10.29.32.0`	Displays all the created variables. The arguments are as follows: • *tftp-file* --Location of the new ROMMON image on the TFTP server. The length of the filename is a maximum of 45 characters. • *ip-address* --IP address on the router to connect to the TFTP server. • *ip-subnet-mask* --IP subnet mask of the router. • *default-gateway* --IP address of the gateway of the TFTP server. • *tftp-server* --IP address of the TFTP server from which the image will be downloaded. **Note** This command is not supported on the Cisco ASR 1000 and Cisco 800 series routers.
Step 3	**sync** **Example:** `ROMMON> sync`	Saves the changes to the image.
Step 4	**tftpdnld -u** **Example:** `ROMMON> tftpdnld -u`	Downloads the new ROMMON image from the TFTP server. • Reset if prompted.

	Command or Action	Purpose
Step 5	**boot** **Example:** `ROMMON> boot`	Boots the router with the Cisco IOS image in flash memory.

Verifying the Upgraded ROMMON Version

To verify that you have an upgraded version of ROMMON, use the show version command:

```
Router# show version
Cisco IOS Software, C828 Software (C828-K9OS&6-M), Version 12.3 (20040702:094716)
[userid 168]
Copyright (c) 1986-2004 by Cisco Systems, Inc.
ROM: System Bootstrap, Version 12.2(11)YV1, Release Software (fc1)
Router uptime is 22 minutes
System returned to ROM by reload
.
.
.
```

Enabling No Service Password-Recovery

Note As a precaution, a valid Cisco IOS image should reside in the bootflash before this feature is enabled.

If you plan to enter the **no service password-recovery** command, Cisco recommends that you save a copy of the system configuration file in a location away from the device.

If you are using a switch that is operating in VLAN Trunking Protocol (VTP) transparent mode, Cisco recommends that you also save a copy of the vlan.dat file in a location away from the switch.

Before You Begin

Always disable the feature before downgrading to an image that does not support this feature, because you cannot reset after the downgrade.

The configuration register boot bit must be enabled so that there is no way to break into ROMMON when this command is configured. Cisco IOS software should prevent the user from configuring the boot field in the config register.

Bit 6, which ignores the startup configuration, and bit 8, which enables a break, should be set.

The Break key should be disabled while the router is booting up and disabled in Cisco IOS software when this feature is enabled.

Perform this task to enable the No Service Password-Recovery feature:

SUMMARY STEPS

1. **enable**
2. **show version**
3. **configure terminal**
4. **config-register** *value*
5. **no service password-recovery**
6. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**show version** **Example:** `Router# show version`	Displays information about the system software, including configuration register settings. • The configuration register must be set to autoboot before entering the **no service password-recovery** command.
Step 3	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 4	**config-register** *value* **Example:** `Router(config)# config-register 0x2102`	(Optional) Changes the configuration register setting. • If necessary, change the configuration register setting so the router is set to autoboot.
Step 5	**no service password-recovery** **Example:** `Router(config)# no service password-recovery`	Disables password-recovery capability at the system console.
Step 6	**exit** **Example:** `Router(config)# exit`	Exits global configuration mode and returns to EXEC mode.

Recovering a Device from the No Service Password-Recovery Feature

To recover a device once the No Service Password-Recovery feature has been enabled, use the telnet client only (SSH is not supported). This telnet client is capable of breaking out of the current session by pressing “Ctrl+]” keys together to arrive at the “telnet>” prompt. At this prompt, you can send multiple strings of “send break” during the “#####” boot up window followed by the “Self decompressing the image :” message. You must send five to seven strings of “send break” during the break out session to arrive at the “Do you want to reset the router to factory default configuration and proceed [y/n] ?” prompt. Answer “yes” for the device to boot with the factory default configuration. If Putty client is used for the telnet session, use the **break** command from the drop down menu. For SecureCRT client with telnet session, press the “Ctrl +break” keys on the full keyboard. The router will not go into ROMmon mode during this recovering process.

If you do not confirm the Break key action, the router boots normally with the No Service Password-Recovery feature enabled.

Examples

This section provides the following examples of the process:

Confirmed Break

```
PASSWORD RECOVERY FUNCTIONALITY IS DISABLED
program load complete, entry point: 0x80013000, size: 0x8396a8
Self decompressing the image :
################################################################
################################################################ [OK]
telnet> send break
telnet> send break
telnet> send break
Restricted Rights Legend
Use, duplication, or disclosure by the Government is subject to restrictions as set forth
in subparagraph (c) of the Commercial Computer Software - Restricted Rights clause at FAR
sec. 52.227-19 and subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer
Software clause at DFARS sec. 252.227-7013.
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, California 95134-1706
Cisco Internetwork Operating System Software
IOS (tm) C2600 Software (C2600-IPBASE-M), Version 12.3(26), RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2008 by cisco Systems, Inc.
Compiled Mon 17-Mar-08 15:24 by dchih
PASSWORD RECOVERY IS DISABLED.
Do you want to reset the router to factory default configuration and proceed [y/n] ?
Reset router configuration to factory default.
This product contains cryptographic features and is subject to United States and local
country laws governing import, export, transfer and use. Delivery of Cisco cryptographic
products does not imply third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for compliance with U.S. and
local country laws. By using this product you agree to comply with applicable laws and
regulations. If you are unable to comply with U.S. and local laws, return this product
immediately.
A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html
If you require further assistance please contact us by sending email to export@cisco.com.
Cisco C831 (MPC857DSL) processor (revision 0x00) with 46695K/2457K bytes of memory.
Processor board ID 0000 (1314672220), with hardware revision 0000 CPU rev number 7
3 Ethernet interfaces
4 FastEthernet interfaces
128K bytes of NVRAM.
24576K bytes of processor board System flash (Read/Write)
```

```
2048K bytes of processor board Web flash (Read/Write)
         --- System Configuration Dialog ---
Would you like to enter the initial configuration dialog? [yes/no]: no
!Start up configuration is erased.
SETUP: new interface FastEthernet1 placed in “up” state
SETUP: new interface FastEthernet2 placed in “up” state
SETUP: new interface FastEthernet3 placed in “up” state
SETUP: new interface FastEthernet4 placed in “up” state
Press RETURN to get started!
Router>
Router> enable
Router# show startup configuration
startup-config is not present
Router# show running-conf ig | in
cl service
no service pad
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!The “no service password-recovery” is disabled.
```

Unconfirmed Break

```
PASSWORD RECOVERY FUNCTIONALITY IS DISABLED
telnet> send break
program load complete, entry point: 0x80013000, size: 0x8396a8
Self decompressing the image :
################################################################################ [OK]
telnet> send break
telnet> send break
Restricted Rights Legend
Use, duplication, or disclosure by the Government is subject to restrictions as set forth
in subparagraph (c) of the Commercial Computer Software - Restricted Rights clause at FAR
sec. 52.227-19 and subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer
Software clause at DFARS sec. 252.227-7013.
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, California 95134-1706
Cisco Internetwork Operating System Software
IOS (tm) C2600 Software (C2600-IPBASE-M), Version 12.3(26), RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2008 by cisco Systems, Inc.
Compiled Mon 17-Mar-08 15:24 by dchih
PASSWORD RECOVERY IS DISABLED.
Do you want to reset the router to factory default configuration and proceed [y/n] ?
PASSWORD RECOVERY IS DISABLED.
Do you want to reset the router to factory default configuration and proceed [y/n] ?
!The user enters “N” here.
This product contains cryptographic features and is subject to United States and local
country laws governing import, export, transfer and use. Delivery of Cisco cryptographic
products does not imply third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for compliance with U.S. and
local country laws. By using this product you agree to comply with applicable laws and
regulations. If you are unable to comply with U.S. and local laws, return this product
immediately.
A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html
If you require further assistance please contact us by sending email to export@cisco.com.
Cisco C831 (MPC857DSL) processor (revision 0x00) with 46695K/2457K bytes of memory.
Processor board ID 0000 (1314672220), with hardware revision 0000
CPU rev number 7
3 Ethernet interfaces
4 FastEthernet interfaces
128K bytes of NVRAM.
24576K bytes of processor board System flash (Read/Write)
2048K bytes of processor board Web flash (Read/Write)
Press RETURN to get started!
!The Cisco IOS software boots as if it is not interrupted.
Router> enable
```

```
Router#
Router# show startup config
Using 984 out of 131072 bytes
!
version 12.3
no service pad
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
no service password-recovery
!
hostname Router
!
boot-start-marker
boot-end-marker
!
memory-size iomem 5
!
no aaa new-model
ip subnet-zero
!
ip ips po max-events 100
no ftp-server write-enable
!
interface Ethernet0
 no ip address
 shutdown
!
interface Ethernet1
 no ip address
 shutdown
 duplex auto
!
interface Ethernet2
 no ip address
 shutdown
!
interface FastEthernet1
 no ip address
 duplex auto
 speed auto
!
interface FastEthernet2
 no ip address
 duplex auto
 speed auto
!
interface FastEthernet3
 no ip address
 duplex auto
 speed auto
!
interface FastEthernet4
 no ip address
 duplex auto
 speed auto
!
ip classless
!
ip http server
no ip http secure-server
!
control-plane
!
line con 0
 no modem enable
 transport preferred all
 transport output all
line aux 0
line vty 0 4
!
scheduler max-task-time 5000
end
```

```
Router# show running-config | incl service
no service pad
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
no service password-recovery
end
```

Configuration Examples for No Service Password-Recovery

Disabling Password Recovery Example

The following example shows how to obtain the configuration register setting (which is set to autoboot), disable password recovery capability, and then verify that the configuration persists through a system reload:

```
Router# show version
Cisco Internetwork Operating System Software
IOS (tm) 5300 Software (C7200-P-M), Version 12.3(8)YA, RELEASE SOFTWARE (fc1)
TAC Support: http://www.cisco.com/tac
Copyright (c) 1986-2004 by Cisco Systems, Inc.
Compiled Wed 05-Mar-04 10:16 by xxx
Image text-base: 0x60008954, data-base: 0x61964000
ROM: System Bootstrap, Version 12.3(8)YA , RELEASE SOFTWARE (fc1)
.
.
.
125440K bytes of ATA PCMCIA card at slot 0 (Sector size 512 bytes).
8192K bytes of Flash internal SIMM (Sector size 256K).
Configuration register is 0x2102
Router# configure terminal
Router(config)# no service password-recovery
WARNING:
Executing this command will disable the password recovery mechanism.
Do not execute this command without another plan for password recovery.
Are you sure you want to continue? [yes]: yes
.
.
.
Router(config)# exit
Router#
Router# reload
Proceed with reload? [confirm] yes
00:01:54: %SYS-5-RELOAD: Reload requested
System Bootstrap, Version 12.3...
Copyright (c) 1994-2004 by cisco Systems, Inc.
C7400 platform with 262144 Kbytes of main memory
PASSWORD RECOVERY FUNCTIONALITY IS DISABLED
.
.
.
```

Additional References

The following sections provide references related to the No Service Password-Recovery feature.

Related Documents

Related Topic	Document Title
Setting, changing, and recovering lost passwords	" Configuring Security with Passwords, Privilege Levels, and Login Usernames for CLI Sessions on Networking Devices " feature module
Loading system images and rebooting	" Using the Cisco IOS Integrated File System " feature module
Security commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	*Cisco IOS Security Command Reference*
Cisco IOS commands	Cisco IOS Master Commands List, All Releases

Standards

Standards	Title
None	--

MIBs

MIBs	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature.	--

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

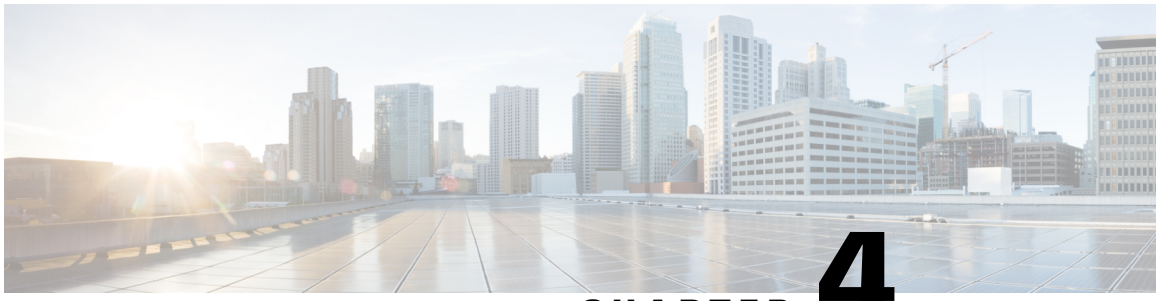
Feature Information for No Service Password-Recovery

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 3: Feature Information for No Service Password-Recovery

Feature Name	Releases	Feature Information
No Service Password-Recovery	12.3(8)YA 12.3(14)T 15.1(1)SY Cisco IOS XE Release 3.10	The No Service Password-Recovery feature is a security enhancement that prevents anyone with console access from accessing the router configuration and clearing the password. It also prevents anyone from changing the configuration register values and accessing NVRAM. This feature was introduced in Cisco IOS Release 12.3(8)YA. This feature was integrated into Cisco IOS Release 12.3(14)T. This feature was integrated into Cisco IOS Release 15.1(1)SY. The following command was introduced: **service password-recovery**. This feature was integrated into Cisco IOS XE Release 3.10 for Cisco ASR 1000 Series Aggregation Services Routers. The following command was modified: **service password-recovery**. The **strict** keyword was added to the **no** form of this command.



CHAPTER 4

Cisco IOS Resilient Configuration

The Cisco IOS Resilient Configuration feature enables a router to secure and maintain a working copy of the running image and configuration so that those files can withstand malicious attempts to erase the contents of persistent storage (NVRAM and flash).

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for Cisco IOS Resilient Configuration

- This feature is available only on platforms that support a Personal Computer Memory Card International Association (PCMCIA) Advanced Technology Attachment (ATA) disk. There must be enough space on the storage device to accommodate at least one Cisco IOS image (two for upgrades) and a copy of the running configuration. IOS Files System (IFS) support for secure file systems is also needed by the software.
- It may be possible to force removal of secured files using an older version of Cisco IOS software that does not contain file system support for hidden files.

- This feature can be disabled only by using a console connection to the router. With the exception of the upgrade scenario, feature activation does not require console access.
- You cannot secure a bootset with an image loaded from the network. The running image must be loaded from persistent storage to be secured as primary.
- Secured files will not appear on the output of a **dir** command issued from an executive shell because the IFS prevents secure files in a directory from being listed. ROM monitor (ROMMON) mode does not have any such restriction and can be used to list and boot secured files. The running image and running configuration archives will not be visible in the Cisco IOS **dir** command output. Instead, use the **show secure bootset** command to verify archive existence.

Information About Cisco IOS Resilient Configuration

Feature Design of Cisco IOS Resilient Configuration

A great challenge of network operators is the total downtime experienced after a router has been compromised and its operating software and configuration data erased from its persistent storage. The operator must retrieve an archived copy (if any) of the configuration and a working image to restore the router. Recovery must then be performed for each affected router, adding to the total network downtime.

The Cisco IOS Resilient Configuration feature is intended to speed up the recovery process. The feature maintains a secure working copy of the router image and the startup configuration at all times. These secure files cannot be removed by the user. This set of image and router running configuration is referred to as the primary bootset.

The following factors were considered in the design of Cisco IOS Resilient Configuration:

- The configuration file in the primary bootset is a copy of the running configuration that was in the router when the feature was first enabled.
- The feature secures the smallest working set of files to preserve persistent storage space. No extra space is required to secure the primary Cisco IOS image file.
- The feature automatically detects image or configuration version mismatch.
- Only local storage is used for securing files, eliminating scalability maintenance challenges from storing multiple images and configurations on TFTP servers.
- The feature can be disabled only through a console session.

How to Use Cisco IOS Resilient Configuration

Archiving a Router Configuration

This task describes how to save a primary bootset to a secure archive in persistent storage.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **secure boot-image**
4. **secure boot-config**
5. **end**
6. **show secure bootset**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**secure boot-image** **Example:** `Router(config)# secure boot-image`	Enables Cisco IOS image resilience.
Step 4	**secure boot-config** **Example:** `Router(config)# secure boot-config`	Stores a secure copy of the primary bootset in persistent storage.
Step 5	**end** **Example:** `Router(config)# end`	Exits to privileged EXEC mode.
Step 6	**show secure bootset** **Example:** `Router# show secure bootset`	(Optional) Displays the status of configuration resilience and the primary bootset filename.

Example

The following example displays sample output from the **show secure bootset** command:

```
Router# show secure bootset
IOS resilience router id JMX0704L5GH
IOS image resilience version 12.3 activated at 08:16:51 UTC Sun Jun 16 2002
Secure archive slot0:c3745-js2-mz type is image (elf) []
  file size is 25469248 bytes, run size is 25634900 bytes
  Runnable image, entry point 0x80008000, run from ram
IOS configuration resilience version 12.3 activated at 08:17:02 UTC Sun Jun 16 2002
Secure archive slot0:.runcfg-20020616-081702.ar type is config
configuration archive size 1059 bytes
```

Restoring an Archived Router Configuration

This task describes how to restore a primary bootset from a secure archive after the router has been tampered with (by an NVRAM erase or a disk format).

Note To restore an archived primary bootset, Cisco IOS image resilience must have been enabled and a primary bootset previously archived in persistent storage.

SUMMARY STEPS

1. **reload**
2. **dir** [*filesystem* **:**]
3. **boot** [*partition-number* **:**][*filename*]
4. **no**
5. **enable**
6. **configure terminal**
7. **secure boot-config** [**restore** *filename*
8. **end**
9. **copy** *filename* **running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**reload** **Example:** `Router# reload`	(Optional) Enters ROM monitor mode, if necessary.
Step 2	**dir** [*filesystem* **:**] **Example:** `rommon 1 > dir slot0:`	Lists the contents of the device that contains the secure bootset file. • The device name can be found in the output of the **show secure bootset** command.

	Command or Action	Purpose
Step 3	**boot** [*partition-number* **:**][*filename*] **Example:** `rommon 2 > boot slot0:c3745-js2-mz`	Boots up the router using the secure bootset image.

Step 4	**no**
	
	Example:
	
	`--- System Configuration Dialog ---`
	
	Example:
	
	`Would you like to enter the initial configuration dialog? [yes/no]: no`
		(Optional) Declines to enter an interactive configuration session in setup mode.
		
		• If the NVRAM was erased, the router enters setup mode and prompts the user to initiate an interactive configuration session.

Step 5	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 6	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 7	**secure boot-config** [**restore** *filename* **Example:** `Router(config)# secure boot-config restore slot0:rescue-cfg`	Restores the secure configuration to the supplied filename.
Step 8	**end** **Example:** `Router(config)# end`	Exits to privileged EXEC mode.
Step 9	**copy** *filename* **running-config** **Example:** `Router# copy slot0:rescue-cfg running-config`	Copies the restored configuration to the running configuration.

Additional References

The following sections provide references related to Cisco IOS Resilient Configuration.

Related Documents

Related Topic	Document Title
Additional commands: complete command syntax, command mode, defaults, usage guidelines, and examples	*The Cisco IOS Configuration Fundamentals and Network Management Command Reference , Release 12.4T*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	--

MIBs

MIBs	MIBs Link
No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	--

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport

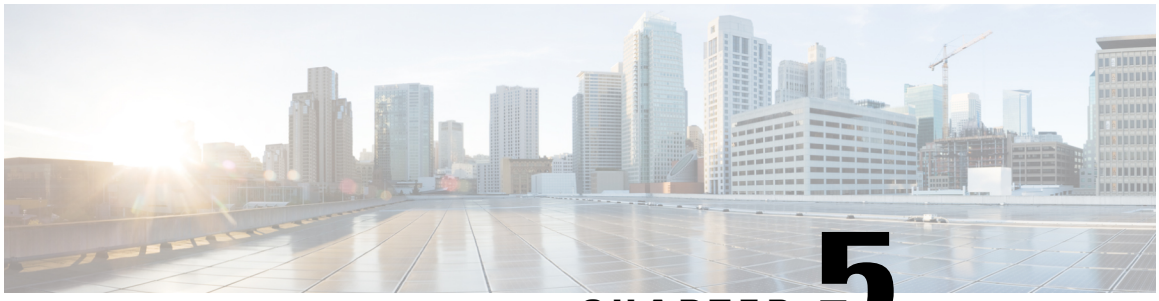
Feature Information for Cisco IOS Resilient Configuration

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 4: Feature Information for Cisco IOS Resilient Configuration

Feature Name	Releases	Feature Information
Cisco IOS Resilient Configuration	12.3(8)T	The Cisco IOS Resilient Configuration feature enables a router to secure and maintain a working copy of the running image and configuration so that those files can withstand malicious attempts to erase the contents of persistent storage (NVRAM and flash). In 12.3(8)T this feature was introduced. The following commands were introduced or modified: **secure boot-config, secure boot-image, show secure bootset.**



CHAPTER 5

Image Verification

The Image Verification feature allows users to automatically verify the integrity of Cisco IOS images. Thus, users can be sure that the image is protected from accidental corruption, which can occur at any time during transit, starting from the moment the files are generated by Cisco until they reach the user. The efficiency of Cisco IOS routers is also improved because the routers can now automatically detect when the integrity of an image is accidentally corrupted as a result of transmission errors or disk corruption.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for Image Verification

Cisco IOS Release 12.2(18)S and 12.0(26)S Only

Image Verification is applied to and attempted on any file; however, if the file is not an image file, image verification will not occur and you will see the following error, “SIGNATURE-NOT-FOUND.”

Cisco IOS Release 12.3(4)T Only

Image Verification is applied only to image files. If any other file type is copied or verified, you will not receive a warning that image verification did occur, and the command (copy or verify) will silently succeed.

Note The Image Verification feature can only be used to check the integrity of a Cisco IOS software image that is stored on a Cisco IOS device. It cannot be used to check the integrity of an image on a remote file system or an image running in memory.

Information About Image Verification

How Image Verification Works

Because a production image undergoes a sequence of transfers before it is copied into the memory of a router, the integrity of the image is at risk of accidental corruption every time a transfer occurs. When downloading an image from Cisco.com, a user can run a message-digest5 (MD5) hash on the downloaded image and verify that the MD5 digest posted on Cisco.com is the same as the MD5 digest that is computed on the user's server. However, many users choose not to run an MD5 digest because it is 128-bits long and the verification is manual. Image verification using SHA-512 digital signature allows the user to automatically validate the integrity of all downloaded images, thereby, significantly reducing user interaction.

How to Use Image Verification

Globally Verifying the Integrity of an Image

The **file verify auto** command enables image verification globally; that is, all images that are to be copied (via the **copy** command) or reloaded (via the **reload** command) are automatically verified. Although both the **copy** and **reload** commands have a **/verify** keyword that enables image verification, you must issue the keyword each time you want to copy or reload an image. The **file verify auto** command enables image verification by default, so you no longer have to specify image verification multiple times.

If you have enabled image verification by default but prefer to disable verification for a specific image copy or reload, the **/noverify** keyword, along with either the **copy** or the **reload** command, will override the **file verify auto** command.

Use this task to enable automatic image verification.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **file verify auto**
4. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**file verify auto** **Example:** `Device(config)# file verify auto`	Enables automatic image verification.
Step 4	**exit** **Example:** `Device(config)# exit`	Exits global configuration mode. You must exit global configuration mode if you are going to copy or reload an image.

What to Do Next

After issuing the **file verify auto** command, you do not have to issue the **/verify** keyword with the **copy** or the **reload** command because each image that is copied or reloaded will be automatically verified.

Verifying the Integrity of an Image That Is About to Be Copied

When issuing the **copy** command, you can verify the integrity of the copied file by entering the **/verify** keyword. If the integrity check fails, the copied file will be deleted. If the file that is about to be copied does not have an embedded hash (an old image), you will be prompted whether or not to continue with the copying process. If you choose to continue, the file will be successfully copied; if you choose not to continue, the copied file will be deleted.

Without the **/verify** keyword, the **copy** command could copy a file that is not valid. Thus, after the **copy** command has been successfully executed, you can issue the **verify** command at any time to check the integrity of the files that are in the storage of the router.

Use this task to verify the integrity of an image before it is copied onto a router.

SUMMARY STEPS

1. **enable**
2. **copy** [**/erase**] [**/verify**| **/noverify**] *source-url destination-url*
3. **verify** [**/md5** [*md5-value*]] *filesystem: file-url*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**copy** [**/erase**] [**/verify**| **/noverify**] *source-url destination-url* **Example:** `Router# copy /verify tftp://10.1.1.1/jdoe/c7200-js-mz disk0:`	Copies any file from a source to a destination. • **/verify** --Verifies the signature of the destination file. If verification fails, the file will be deleted. • **/noverify** --Does not verify the signature of the destination file before the image is copied. **Note** **/noverify** is often issued if the **file verify auto** command is enabled, which automatically verifies the signature of all images that are copied.
Step 3	**verify** [**/md5** [*md5-value*]] *filesystem: file-url*] **Example:** `Router# verify bootflash://c7200-kboot-mz.121-8a.E`	(Optional) Verifies the integrity of the images in the router's storage.

Verifying the Integrity of an Image That Is About to Be Reloaded

By issuing the **reload** command with the /**verify** keyword, the image that is about to be loaded onto your system will be checked for integrity. If the /**verify** keyword is specified, image verification will occur before the system initiates the reboot. Thus, if verification fails, the image will not be loaded.

Note Because different platforms obtain the file that is to be loaded in various ways, the file specified in BOOTVAR will be verified. If a file is not specified, the first file on each subsystem will be verified. On certain platforms, because of variables such as the configuration register, the file that is verified may not be the file that is loaded.

Use this task to verify the integrity of an image before it is reloaded onto a router.

SUMMARY STEPS

1. **enable**
2. **reload** [[**warm**] [**/verify**| **/noverify**] *text* | [**warm**] [**/verify**| **/noverify**] **in** [*hh* **:** *mm* [*text*] | [**warm**] [**/verify**| **/noverify**] **at** *hh* **:** *mm* [*month day* | *day month*] [*text*] | [**warm**] [**/verify**| **/noverify**] **cancel**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**reload** [[**warm**] [**/verify**| **/noverify**] *text*	[**warm**] [**/verify**| **/noverify**] **in** [*hh* **:** *mm* [*text*]	[**warm**] [**/verify**| **/noverify**] **at** *hh* **:** *mm* [*month day*	*day month*] [*text*]	[**warm**] [**/verify**| **/noverify**] **cancel**] **Example:** `Router# reload /verify`	Reloads the operating system. • /**verify**--Verifies the signature of the destination file. If verification fails, the file will be deleted. • **/noverify** --Does not verify the signature of the destination file before the image is reloaded. **Note** **/noverify** is often issued if the **file verify auto** command is enabled, which automatically verifies the signature of all images that are copied.

Configuration Examples for Image Verification

Global Image Verification Example

The following example shows how to enable automatic image verification. After enabling this command, image verification will automatically occur for all images that are either copied (via the **copy** command) or reloaded (via the **reload** command).

```
Device(config)# file verify auto
```

Image Verification via the copy Command Example

The following example shows how to specify image verification before copying an image:

```
Device# copy /verify tftp://10.1.1.1/jdoe/c7200-js-mz disk0:
Destination filename [c7200-js-mz]?
Accessing tftp://10.1.1.1/jdoe/c7200-js-mz...
Loading jdoe/c7200-js-mz from 10.1.1.1 (via FastEthernet0/0):!!!!!!!!!!!!!!!
```

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
[OK - 19879944 bytes]
19879944 bytes copied in 108.632 secs (183003 bytes/sec)
Verifying file integrity of disk0:/c7200-js-mz
.....................................................................................
.....................................................................................
.....................................................................................
.........................Done!
Embedded Hash   MD5 :CFA258948C4ECE52085DCF428A426DCD
Computed Hash       MD5 :CFA258948C4ECE52085DCF428A426DCD
CCO Hash               MD5 :44A7B9BDDD9638128C35528466318183
Signature Verified
```

Image Verification via the reload Command Example

The following example shows how to specify image verification before reloading an image onto the Device:

```
Device# reload /verify
Verifying file integrity of bootflash:c7200-kboot-mz.121-8a.E
%ERROR:Signature not found in file bootflash:c7200-kboot-mz.121-8a.E.
Signature not present. Proceed with verify? [confirm]
Verifying file disk0:c7200-js-mz
.....................................................................
.....................................................Done!
Embedded Hash     MD5 :CFA258948C4ECE52085DCF428A426DCD
Computed Hash      MD5 :CFA258948C4ECE52085DCF428A426DCD
CCO Hash          MD5 :44A7B9BDDD9638128C35528466318183
Signature Verified
Proceed with reload? [confirm]n
```

Verify Command Sample Output Example

The following example shows how to specify image verification via the **verify** command:

```
Device# verify disk0:c7200-js-mz
%Filesystem does not support verify operations
Verifying file integrity of disk0:c7200-js-mz.....................................
.....................................................................................Done!
Embedded Hash     MD5 :CFA258948C4ECE52085DCF428A426DCD
Computed Hash      MD5 :CFA258948C4ECE52085DCF428A426DCD
CCO Hash          MD5 :44A7B9BDDD9638128C35528466318183
Signature Verified
```

Additional References

Related Documents

Related Topic	Document Title
Configuration tasks and information for loading, maintaining, and rebooting system images	Using the Cisco IOS Integrated File System feature module in the *Cisco IOS Configuration Fundamentals and Network Management Configuration Guide* , Release 12.4T.

Related Topic	Document Title
Additional commands for loading, maintaining, and rebooting system images	*Cisco IOS Configuration Fundamentals Command Reference* , Release 12.4T

Standards

Standard	Title
None	--

MIBs

MIB	MIBs Link
• None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
None	--

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport

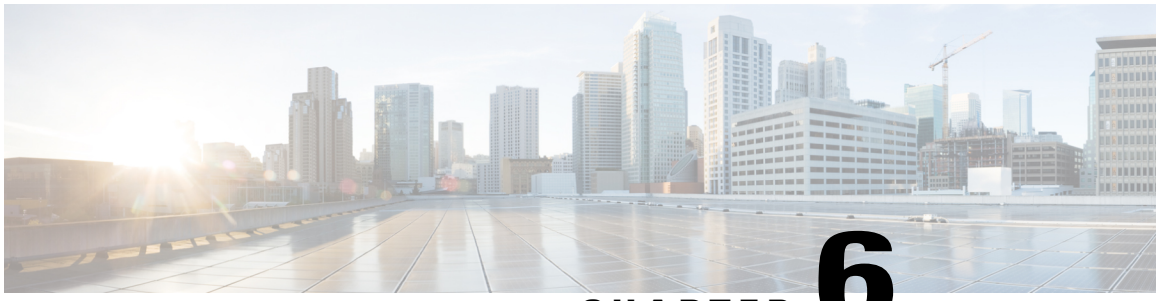
Feature Information for Image Verification

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 5: Feature Information for Image Verification

Feature Name	Releases	Feature Information
Image Verification	Cisco IOS 12.2(25)S Cisco IOS 12.0(26)S Cisco IOS 12.3(4)T	The Image Verification feature allows users to automatically verify the integrity of Cisco IOS images. The following commands were introduced or modified: **copy**, **file verify auto**, **reload**, **verify**.



CHAPTER 6

IP Source Tracker

The IP Source Tracker feature tracks information in the following ways:

- Gathers information about the traffic that is flowing to a host that is suspected of being under attack.
- Generates all the necessary information in an easy-to-use format to track the network entry point of a DoS attack.
- Tracks Multiple IPs at the same time.
- Tracks DoS attacks across the entire network.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for IP Source Tracker

Packets Can Be Dropped for Routers

IP source tracking is designed to track attacks against hosts. Packets can be dropped if the line card or port adapter CPU is overwhelmed. Therefore, when used to track an attack against a router, IP source tracking can drop control packets, such as Border Gateway Protocol (BGP) updates.

Engine 0 and 1 Performances Affected on Cisco 12000 Series

There is no performance impact for packets destined to nontracked IP addresses on Engine 2 and Engine 4 line cards because the IP source tracker affects only tracked destinations. Engine 0 and Engine 1 performances are affected because on these engines all packets are switched by the CPU.

Note On Cisco 7500 series routers, there is no performance impact on destinations that are not tracked.

Information About IP Source Tracker

Identifying and Tracking Denial of Service Attacks

One of the many challenges faced by customers today is the tracking and blocking denial-of-service (DoS) attacks. Counteracting a DoS attack involves intrusion detection, source tracking, and blocking. This functionality addresses the need for source tracking.

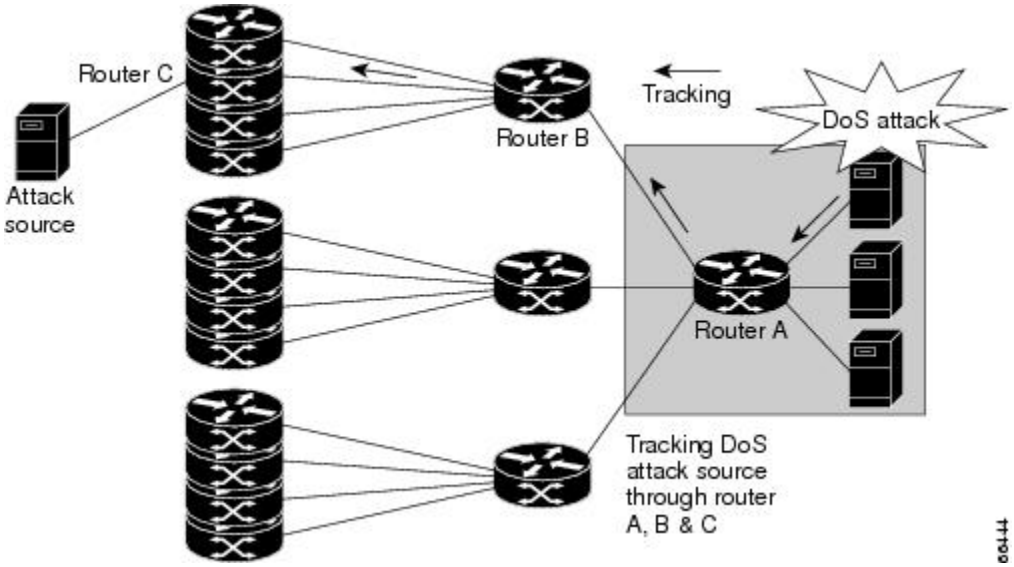
To trace attacks, NetFlow and access control lists (ACLs) have been used. To block attacks, committed access rate (CAR) and ACLs have been used. Support for these features on the Cisco 12000 series Internet router has depended on the type of line card used. Support for these features on the Cisco 7500 series routers depends upon the type of port adapter used. There is, therefore, a need to develop a way to receive information that both traces the source of an attack and is supported on all line cards and port adapters.

Normally, when you identify the host that is subject to a DoS attack, you must determine the network ingress point to effectively block the attack. This process starts at the router closest to the host.

For example, in the figure below, you would start at Router A and try to determine the next upstream router to examine. Traditionally, you would apply an output ACL to the interface connecting to the host to log packets that match the ACL. The logging information is dumped to the router console or system log. You then have to analyze this information, and possibly go through several ACLs in succession to identify the input interface for the attack. In this case the information points back to Router B.

You then repeat this process on Router B, which leads back to Router C, an ingress point into the network. At this point you can use ACLs or CAR to block the attack. This procedure can require applying several ACLs that generate an excessive amount of output to analyze, making this procedure cumbersome and error prone.

Figure 1: Source Tracking in a DoS Attack



Using IP Source Tracker

IP source tracker provides an easier, more scalable alternative to output ACLs for tracking DoS attacks, and it works as follows:

- After you identify the destination being attacked, enable tracking for the destination address on the whole router by entering the **ip source-track** command.
- Each line card creates a special Cisco Express Forwarding (CEF) entry for the destination address being tracked. For line cards or port adapters that use specialized Application-Specific Integrated Circuit (ASICs) for packet switching, the CEF entry is used to punt packets to the line card's or port adapter's CPU.
- Each line card CPU collects information about the traffic flow to the tracked destination.
- The data generated is periodically exported to the router. To display a summary of the flow information, enter the **show ip source-track summary** command. To display more detailed information for each input interface, enter the **show ip source-track** command.
- Statistics provide a breakdown of the traffic to each tracked IP address. This breakdown allows you to determine which upstream router to analyze next. You can shut down the IP source tracker on the current router by entering the **no ip source-track** command, and reopen it on the upstream router.
- Repeat Step 1 to Step 5 until you identify the source of the attack.
- Apply CAR or ACLs to limit or stop the attack.

IP Source Tracker Hardware Support

IP source tracking is supported on all Engine 0, 1, 2, and 4 line cards in the Cisco 12000 series Internet router. It is also supported on all port adapters and RSPs that have CEF switching enabled on Cisco 7500 series routers.

How to Configure IP Source Tracker

Configuring IP Source Tracking

To configure IP source tracking for a host under attack, perform the following steps.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip source-track** *ip-address*
4. **ip source-track address-limit** *number*
5. **ip source-track syslog-interval** *number*
6. **ip source-track export-interval** *number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip source-track** *ip-address* **Example:** `Router(config)# ip source-track 100.10.0.1`	Enables IP source tracking for a specified host.

	Command or Action	Purpose
Step 4	**ip source-track address-limit** *number* **Example:** `Router(config)# ip source-track address-limit 10`	(Optional) Limits the number of hosts that can be simultaneously tracked at any given time. **Note** If this command is not enabled, there is no limit to the number of hosts that be can tracked.
Step 5	**ip source-track syslog-interval** *number* **Example:** `Router(config)# ip source-track syslog-interval 2`	(Optional) Sets the time interval, in minutes, used to generate syslog messages that indicate IP source tracking is enabled. **Note** If this command is not enabled, system log messages are not generated.
Step 6	**ip source-track export-interval** *number* **Example:** `Router(config)# ip source-track export-interval 30`	(Optional) Sets the time interval, in seconds, used to export IP tracking statistics that are collected in the line cards to the gigabit route processor (GRP) and the port adapters to the route switch processor (RSP). **Note** If this command is not enabled, traffic flow information is exported to the GRP and RSP every 30 seconds.

What to Do Next

After you have configured source tracking on your network device, you can verify your configuration and source tracking statistics, such as traffic flow. To complete this task, see the following section "Verifying IP Source Tracking, on page 81."

Verifying IP Source Tracking

To verify the status of source tracking, such as packet processing and traffic flow information, perform the following steps.

SUMMARY STEPS

1. **enable**
2. **show ip source-track** [*ip-address*] [**summary** | **cache**
3. **show ip source-track export flows**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: `Router> enable`	• Enter your password if prompted.
Step 2	**show ip source-track** [*ip-address*] [**summary** | **cache** **Example:** `Router# show ip source-track summary`	Displays traffic flow statistics for tracked IP host addresses
Step 3	**show ip source-track export flows** **Example:** `Router# show ip source-track export flows`	Displays the last 10 packet flows that were exported from the line card to the route processor. **Note** This command can be issued only on distributed platforms, such as the GRP and the RSP.

Example

The following example, which is sample output from the **show ip source-track summary** command, shows how to verify that IP source tracking is enabled for one or more hosts:

```
Router# show ip source-track summary
Address          Bytes    Pkts    Bytes/s   Pkts/s
10.0.0.1          119G   1194M    443535      4432
192.168.1.1       119G   1194M    443535      4432
192.168.42.42     119G   1194M    443535      4432
```

The following example, which is sample output from the **show ip source-track summary** command, shows how to verify that no traffic has yet to be received for the destination hosts that are being tracked:

```
Router# show ip source-track summary
Address        Bytes   Pkts   Bytes/s   Pkts/s
10.0.0.1           0      0         0        0
192.168.1.1        0      0         0        0
192.168.42.42      0      0         0        0
```

The following example, which is sample output from the **show ip source-track**command, shows how to verify that IP source tracking is processing packets to the hosts and exporting statistics from the line card or port adapter to the GRP and RSP:

```
Router# show ip source-track
Address         SrcIF    Bytes   Pkts   Bytes/s   Pkts/s
10.0.0.1        PO0/0    119G   1194M    513009     5127
192.168.1.1     PO0/0    119G   1194M    513009     5127
```

192.168.42.42 PO0/0 119G 1194M 513009 5127

Configuration Examples for IP Source Tracker

Configuring IP Source Tracking Example

The following example shows how to configure IP source tracking on all line cards and port adapters in the router. In this example, each line card or port adapter collects traffic flow data to host address 100.10.0.1 for 2 minutes before creating an internal system log entry; packet and flow information recorded in the system log is exported for viewing to the route processor or switch processor every 60 seconds.

```
Router# configure interface
Router(config)# ip source-track 100.10.0.1
Router(config)# ip source-track syslog-interval 2
Router(config)# ip source-track export-interval 60
```

Verifying Source Interface Statistics for All Tracked IP Addresses Example

The following example displays a summary of the traffic flow statistics that are collected on each source interface for tracked host addresses.

```
Router# show ip source-track
Address         SrcIF     Bytes    Pkts     Bytes/s     Pkts/s
10.0.0.1        PO2/0         0       0           0          0
192.168.9.9     PO1/2      131M    511M        1538          6
192.168.9.9     PO2/0      144G   3134M     6619923     143909
```

Verifying a Flow Statistic Summary for All Tracked IP Addresses Example

The following example displays a summary of traffic flow statistics for all hosts that are being tracked; it shows that no traffic has yet been received.

```
Router# show ip source-track summary
Address              Bytes    Pkts     Bytes/s     Pkts/s
10.0.0.1                 0       0           0          0
100.10.1.1            131M    511M        1538          6
192.168.9.9           146G   3178M     6711866     145908
```

Verifying Detailed Flow Statistics Collected by a Line Card Example

The following example displays traffic flow information that is collected on line card 0 for all tracked hosts.

```
Router# exec slot 0 show ip source-track cache
========= Line Card (Slot 0) =======
IP packet size distribution (7169M total packets):
   1-32   64   96  128  160  192  224  256  288  320  352  384  416  448  480
   .000 .000 .000 0.00 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000
    512  544  576 1024 1536 2048 2560 3072 3584 4096 4608
   .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000
IP Flow Switching Cache, 278544 bytes
  1 active, 4095 inactive, 13291 added
  198735 ager polls, 0 flow alloc failures
  Active flows timeout in 0 minutes
  Inactive flows timeout in 15 seconds
```

```
  last clearing of statistics never
Protocol         Total    Flows   Packets Bytes  Packets Active(Sec) Idle(Sec)
--------         Flows     /Sec     /Flow  /Pkt     /Sec     /Flow     /Flow
SrcIf          SrcIPaddress    DstIf          DstIPaddress    Pr TOS Flgs  Pkts
Port Msk AS                    Port Msk AS    NextHop              B/Pk  Active
PO0/0          101.1.1.0       Null           100.1.1.1       06 00  00      55K
0000 /0  0                     0000 /0  0     0.0.0.0                100    10.1
```

Verifying Flow Statistics Exported from Line Cards and Port Adapters Example

The following example displays packet flow information that is exported from line cards and port adapters to the GRP and the RSP:

```
Router# show ip source-track export flows
SrcIf         SrcIPaddress    DstIf         DstIPaddress    Pr SrcP DstP  Pkts
PO0/0         101.1.1.0       Null          100.1.1.1       06 0000 0000    88K
PO0/0         101.1.1.0       Null          100.1.1.3       06 0000 0000    88K
PO0/0         101.1.1.0       Null          100.1.1.2       06 0000 0000    88K
```

Additional References

The following sections provide references related to IP Source Tracker.

Related Documents

Related Topic	Document Title
ACLs	*Cisco IOS Security Configuration Guide: Securing the Data Plane* , Release 12.4T
Dynamic ACLs	Configuring Lock-and-Key Security (Dynamic Access Lists)
DoS prevention	Configuring TCP Intercept (Preventing Denial-of-Service Attacks)

Standards

Standards	Title
None	--

MIBs

MIBs	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
None	--

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport

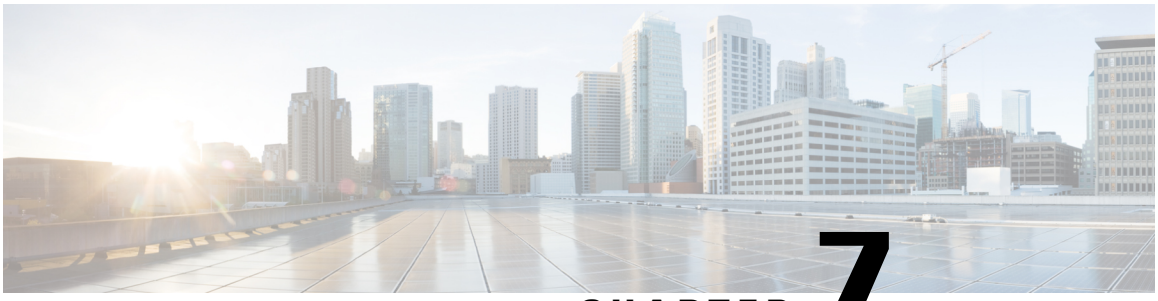
Feature Information for IP Source Tracker

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 6: Feature Information for IP Source Tracker

Feature Name	Releases	Feature Information
IP Source Tracker	12.0(21)S 12.0(22)S 12.0(26)S 12.3(7)T 12.2(25)S	The IP Source Tracker feature allows information to be gathered about the traffic that is flowing to a host that is suspected of being under attack. This feature was introduced in Release 12.0(21)S on the Cisco 12000 series. This feature was implemented in Release 12.0(22)S on the Cisco 7500 series. This feature was implemented in Release 12.0(26)S on the Cisco 12000 series IP Service Engine (ISE) line cards. This feature was integrated into Cisco IOS Release 12.3(7)T. This feature was integrated into Cisco IOS Release 12.2(25)S. The following commands were introduced or modified: **ip source-track, ip source-track address-limit, ip source-track export-interval, ip source-track syslog-interval, show ip source-track, show ip source-track export flows.**



CHAPTER 7

Role-Based CLI Access

The Role-Based CLI Access feature allows the network administrator to define views, which are a set of operational commands and configuration capabilities that provide selective or partial access to Cisco IOS EXEC and configuration (config) mode commands. Views restrict user access to Cisco IOS command-line interface (CLI) and configuration information; that is, a view can define what commands are accepted and what configuration information is visible. Thus, network administrators can exercise better control over access to Cisco networking devices.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Role-Based CLI Access

Your image must support CLI views.

Restrictions for Role-Based CLI Access

Lawful Intercept Images Limitation

CLI views are a part of all platforms and Cisco IOS images because they are a part of the Cisco IOS parser. However, the lawful intercept view is available only in images that contain the lawful intercept subsystem.

Maximum Number of Allowed Views

The maximum number of CLI views and superviews, including one lawful intercept view, that can be configured is 15. (This does not include the root view.)

Information About Role-Based CLI Access

Benefits of Using CLI Views

Although users can control CLI access via both privilege levels and enable mode passwords, these functions do not provide network administrators with the necessary level of detail needed when working with Cisco IOS devices. CLI views provide a more detailed access control capability for network administrators, thereby, improving the overall security and accountability of Cisco IOS software.

As of Cisco IOS Release 12.3(11)T, network administrators can also specify an interface or a group of interfaces to a view; thereby, allowing access on the basis of specified interfaces.

Root View

When a system is in root view, it has all of the access privileges as a user who has level 15 privileges. If the administrator wishes to configure any view to the system (such as a CLI view, a superview, or a lawful intercept view), the system must be in root view.

The difference between a user who has level 15 privileges and a root view user is that a root view user can configure a new view and add or remove commands from the view. Also, when you are in a CLI view, you have access only to the commands that have been added to that view by the root view user.

Lawful Intercept View

Like a CLI view, a lawful intercept view restricts access to specified commands and configuration information. Specifically, a lawful intercept view allows a user to secure access to lawful intercept commands that are held within the TAP-MIB, which is a special set of simple network management protocol (SNMP) commands that store information about calls and users.

Commands available in lawful intercept view belong to one of the these categories:

- Lawful intercept commands that should not be made available to any other view or privilege level
- CLI views that are useful for lawful intercept users but do not have to be excluded from other views or privilege levels

Superview

A superview consists of one or more CLI views, which allow users to define what commands are accepted and what configuration information is visible. Superviews allow a network administrator to easily assign all users within configured CLI views to a superview instead of having to assign multiple CLI views to a group of users.

Superviews contain these characteristics:

- A CLI view can be shared among multiple superviews.
- Commands cannot be configured for a superview; that is, you must add commands to the CLI view and add that CLI view to the superview.
- Users who are logged into a superview can access all of the commands that are configured for any of the CLI views that are part of the superview.
- Each superview has a password that is used to switch between superviews or from a CLI view to a superview.
- If a superview is deleted, its associated CLI views are not deleted.

View Authentication via a New AAA Attribute

View authentication is performed by an external authentication, authorization, and accounting (AAA) server via the new attribute **cli-view-name**.

AAA authentication associates only one view name to a particular user; that is, only one view name can be configured for a user in an authentication server.

How to Use Role-Based CLI Access

Configuring a CLI View

Perform this task to create a CLI view and add commands or interfaces to the view, as appropriate.

Before You Begin

Before you create a view, you must perform the following tasks:

- Enable AAA using the **aaa new-model** command.
- Ensure that your system is in root view-not privilege level 15.

SUMMARY STEPS

1. **enable view**
2. **configure terminal**
3. **parser view** *view-name* [**inclusive**]
4. **secret** [**0** | **5**] *encrypted-password*
5. **commands** *parser-mode* {**exclude** | **include-exclusive** | **include**} [**all**] [**interface** *interface-name* | *command*]
6. **end**
7. **enable** [*privilege-level* | **view** *view-name*]
8. **show parser view all**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable view** **Example:** `Device> enable view`	Enables root view. • Enter your privilege level 15 password (for example, root password) if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

Step 3	**parser view** *view-name* [**inclusive**]
	
	Example:
	
	`Device(config)# parser view first inclusive`
	`Device(config-view)#`
		Creates a view including all commands by default. If the **inclusive** keyword option is not selected, it creates a view excluding all commands by default. You are in the view configuration mode.

Step 4	**secret** [**0**	**5**] *encrypted-password* **Example:** `Device(config-view)# secret 5 secret`	Associates a CLI view or superview with a password. **Note** You must issue this command before you can configure additional attributes for the view. **Note** With CSCts50236, the password can be removed or overwritten. Use the **no secret** command to remove the configured password.
Step 5	**commands** *parser-mode* {**exclude**	**include-exclusive**	**include**} [**all**] [**interface** *interface-name*	*command*]	Adds commands or interfaces to a view and specifies the mode in which the specified command exists.

	Command or Action	Purpose
	Example: `Device(config-view)# commands exec include show version`	**Note** While configuring parser view profiles, the following **no** or **default** commands are not saved to the startup configuration. These commands are in use until the device is reloaded. Once the device is reloaded, reapply these commands to get the required results. • **commands configure include all no** • **commands interface include all no** • **commands configure include all default**
Step 6	**end** **Example:** `Device(config-view)# end`	Exits view configuration mode and returns to privileged EXEC mode.
Step 7	**enable** [*privilege-level* | **view** *view-name*] **Example:** `Device# enable view first`	Prompts you for a password to access a configured CLI view, and you can switch from one view to another view. Enter the password to access the CLI view.
Step 8	**show parser view all** **Example:** `Device# show parser view all`	(Optional) Displays information for all views that are configured on the device. **Note** Although this command is available for both root and lawful intercept users, the **all** keyword is available only to root users. However, the **all** keyword can be configured by a user in root view to be available for users in lawful intercept view and CLI view.

Troubleshooting Tips

You must associate a password with a view. If you do not associate a password, and you attempt to add commands to the view using the **commands** command, a system message such as the following is displayed:

```
%Password not set for view <viewname>.
```

Configuring a Lawful Intercept View

Perform this task to initialize and configure a view for lawful-intercept-specific commands and configuration information.

Before You Begin

Before you initialize a lawful intercept view, ensure that the privilege level is set to 15 using the **privilege** command.

Note Only an administrator or a user who has level 15 privileges can initialize a lawful intercept view.

SUMMARY STEPS

1. **enable view**
2. **configure terminal**
3. **li-view** *li-password* **user** *username* **password** *password*
4. **username lawful-intercept** [*name*] [**privilege** *privilege-level* | **view** *view-name*] **password** *password*
5. **parser view** *view-name*
6. **secret 5** *encrypted-password*
7. **name** *new-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable view** **Example:** `Device> enable view`	Enables root view. • Enter your privilege level 15 password (for example, root password) if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**li-view** *li-password* **user** *username* **password** *password* **Example:** `Device(config)# li-view lipass user li_admin password li_adminpass`	Initializes a lawful intercept view. After the li-view is initialized, you must specify at least one user via **user** *username* **password** *password* options.
Step 4	**username lawful-intercept** [*name*] [**privilege** *privilege-level*	**view** *view-name*] **password** *password* **Example:** `Device(config)# username lawful-intercept li-user1 password li-user1pass`	Configures lawful intercept users on a Cisco device.

	Command or Action	Purpose
Step 5	**parser view** *view-name* **Example:** `Device(config)# parser view li view name`	(Optional) Enters view configuration mode, which allows you to change the lawful intercept view password or the lawful intercept view name.
Step 6	**secret 5** *encrypted-password* **Example:** `Device(config-view)# secret 5 secret`	(Optional) Changes an existing password for a lawful intercept view.
Step 7	**name** *new-name* **Example:** `Device(config-view)# name second`	(Optional) Changes the name of a lawful intercept view. If this command is not issued, the default name of the lawful intercept view is “li-view.”

Troubleshooting Tips

To display information for all users who have access to a lawful intercept view, issue the **show users lawful-intercept** command. (This command is available only to authorized lawful intercept view users.)

Configuring a Superview

Perform this task to create a superview and add at least one CLI view to the superview.

Before You Begin

Before adding a CLI view to a superview, ensure that the CLI views that are added to the superview are valid views in the system; that is, the views have been successfully created using the **parser view** command.

Note You can add a view to a superview only after you configure a password for the superview (using the **secret 5** command). Thereafter, issue the **view** command in view configuration mode to add at least one CLI view to the superview.

SUMMARY STEPS

1. **enable view**
2. **configure terminal**
3. **parser view** *superview-name* **superview**
4. **secret 5** *encrypted-password*
5. **view** *view-name*
6. **end**
7. **show parser view all**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable view** **Example:** `Device> enable view`	Enables root view. • Enter your privilege level 15 password (for example, root password) if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**parser view** *superview-name* **superview** **Example:** `Device(config)# parser view su_view1 superview`	Creates a superview and enters view configuration mode.
Step 4	**secret 5** *encrypted-password* **Example:** `Device(config-view)# secret 5 secret`	Associates a CLI view or superview with a password. **Note** You must issue this command before you can configure additional attributes for the view.
Step 5	**view** *view-name* **Example:** `Device(config-view)# view view_three`	Adds a normal CLI view to a superview. Issue this command for each CLI view that is to be added to a given superview.

Step 6	**end**
	
	Example:
	`Device(config-view)# end`
	`Device#`
		Exits view configuration mode and returns to privileged EXEC mode.

	Command or Action	Purpose
Step 7	**show parser view all** **Example:** `Device# show parser view`	(Optional) Displays information for all views that are configured on the device. **Note** Although this command is available for both root and lawful intercept users, the **all** keyword is available only to root users. However, the **all** keyword can be configured by a user in root view to be available for users in lawful intercept view and CLI view.

Monitoring Views and View Users

To display debug messages for all views-root, CLI, lawful intercept, and superview-use the **debug parser view** command in privileged EXEC mode.

Configuration Examples for Role-Based CLI Access

Example: Configuring a CLI View

The following example shows how to configure two CLI views, “first” and “second". Thereafter, you can verify the CLI view in the running configuration.

```
Device(config)# parser view first inclusive
Device(config-view)# secret 5 firstpass
Device(config-view)# command exec exclude show version
Device(config-view)# command exec exclude configure terminal
Device(config-view)# command exec exclude all show ip
Device(config-view)# exit
Device(config)# parser view second
Device(config-view)# secret 5 secondpass
Device(config-view)# command exec include-exclusive show ip interface
Device(config-view)# command exec include logout
Device(config-view)# exit
!
!
Device(config-view)# do show running-config | beg view

parser view first inclusive
 secret 5 $1$MCmh$QuZaU8PIMPlff9sFCZvgW/
 commands exec exclude configure terminal
 commands exec exclude configure
 commands exec exclude all show ip
 commands exec exclude show version
 commands exec exclude show
!
parser view second
 secret 5 $1$iP2M$R16BXKecMEiQesxLyqygW.
 commands exec include-exclusive show ip interface
 commands exec include show ip
 commands exec include show
 commands exec include logout
!
```

Example: Verifying a CLI View

After you have configured the CLI views “first” and “second", you can issue the **enable view**command to verify which commands are available in each view. The following example shows which commands are available inside the CLI view “first” after the user has logged into this view. (Because the **show ip** command is configured with the all option, a complete set of suboptions is shown, except the **show ip interface** command, which is using the **include-exclusive** keyword in the second view.)

```
Device# enable view first
Password:
Device# ?
Exec commands:
  configure  Enter configuration mode
  enable     Turn on privileged commands
  exit       Exit from the EXEC
  show       Show running system information
Device# show ?
  ip       IP information
  parser   Display parser information
  version  System hardware and software status
Device# show ip ?

  access-lists            List IP access lists
  accounting              The active IP accounting database
  aliases                 IP alias table
  arp                     IP ARP table
  as-path-access-list     List AS path access lists
  bgp                     BGP information
  cache                   IP fast-switching route cache
  casa                    display casa information
  cef                     Cisco Express Forwarding
  community-list          List community-list
  dfp                     DFP information
  dhcp                    Show items in the DHCP database
  drp                     Director response protocol
  dvmrp                   DVMRP information
  eigrp                   IP-EIGRP show commands
  extcommunity-list       List extended-community list
  flow                    NetFlow switching
  helper-address          helper-address table
  http                    HTTP information
  igmp                    IGMP information
  irdp                    ICMP Device Discovery Protocol
.
.
.
```

Example: Configuring a Lawful Intercept View

The following example shows how to configure a lawful intercept view, add users to the view, and verify the users that were added:

```
!Initialize the LI-View.
Device(config)# li-view lipass user li_admin password li_adminpass
Device(config)# end
! Enter the LI-View; that is, check to see what commands are available within the view.
Device# enable view li-view
Password:
Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# parser view li-view

Device(config-view)# ?
```

```
View commands:
  commands  Configure commands for a view
  default   Set a command to its defaults
  exit      Exit from view configuration mode
  name      New LI-View name
  no        Negate a command or set its defaults
  password  Set a password associated with CLI views
Device(config-view)#
! NOTE:LI View configurations are never shown as part of 'running-configuration'.
! Configure LI Users.
Device(config)# username lawful-intercept li-user1 password li-user1pass

Device(config)# username lawful-intercept li-user2 password li-user2pass
! Displaying LI User information.
Device# show users lawful-intercept
li_admin
li-user1
li-user2
Device#
```

Note The lawful intercept view is available only on specific images and the view name option is available only in the LI view.

Example: Configuring a Superview

The following sample output from the **show running-config** command shows that "view_one" and "view_two" have been added to superview "su_view1", "view_three", and "view_four" have been added to superview "su_view2":

```
Device# show running-config
!
parser view su_view1 superview
 secret 5 <encoded password>
 view view_one
 view view_two
!
parser view su_view2 superview
 secret 5 <encoded password>
 view view_three
 view view_four
!
```

Additional References for Role-Based CLI Access

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases

Related Topic	Document Title
Security commands	• Cisco IOS Security Command Reference: Commands A to C • Cisco IOS Security Command Reference: Commands D to L • Cisco IOS Security Command Reference: Commands M to R • Cisco IOS Security Command Reference: Commands S to Z
SNMP, MIBs, CLI configuration	*Cisco IOS Network Management Configuration Guide* , Release 15.0.
Privilege levels	"Configuring Security with Passwords, Privileges and Logins" module.

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Role-Based CLI Access

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 7: Feature Information for Role-Based CLI Access

Feature Name	Releases	Feature Information
Role-Based CLI Access	Cisco IOS 12.3(7)T Cisco IOS 12.3(11)T Cisco IOS 12.2(33)SRB Cisco IOS 12.2(33)SB Cisco IOS 12.2(33)SXI	The Role-Based CLI Access feature enables network administrators to restrict user access to CLI and configuration information. The CLI view capability was extended to restrict user access on a per-interface level, and additional CLI views were introduced to support the extended view capability. Also, support to group configured CLI views into a superview was introduced. The following commands were introduced or modified: **commands (view)**, **enable**, **li-view**, **name (view)**, **parser view , parser view superview**, **secret**, **show parser view**, **show users**, **username**, and **view**.
Role-Based CLI Inclusive Views	Cisco IOS 15.4(1)T Cisco IOS 15.4(1)S Cisco IOS 15.2(1)SY	The Role-Based CLI Inclusive Views feature enables a standard CLI view including all commands by default. The following command was modified: **parser view inclusive**.